

SNO	Text	Tags	Dominant Topic	Topic Terms	URL	Label	No. of Answers (Number)	Is Best Answer Selected? (Yes/No)	Is the question Theoretical or Program related? (Theoretical/Programming)	Is the solution temporary or long term? (Temporary/Long Term)	Is the discussion related to design? (Yes/No)	Is label representative? (Yes/No)
1	properties self argument know define methods class blew away attempt next class called bank class called savings bank account constructor takes argument constructor sets property minimum balance given savings account class implement takes cash deposit amounts updates balance accordingly returns negative deposit amounts return invalid deposit amount savings account class implement method takes cash withdrawal amount deducts amount current balance returns balance method balance below check condition withdraw beyond minimum account happens withdraw beyond current account withdrawal amount greater current balance negative withdrawal amount withdraw class called current bank account constructor takes argument sets property current account class implement takes cash deposit amounts updates balance accordingly returns negative deposit amount return invalid deposit amount current account class implement takes cash withdrawal amount deducts amount current balance returns balance negative withdrawal amount return invalid withdraw amount withdrawing current balance fail message cannot withdraw beyond current account attempt	class-design	1	output, text, document, format, button, copy, account, original, date, click, flow, insert, index, present, target, thought, suggestion, enter, development, enable	https://stackoverflow.com/questions/39322269/how-do-i-set-properties-in-a-constructor-that-has-only-a-self-argument-like-d	Interface design pattern (User Interface, Frameworks)	3	yes	program related	temporary	yes	no
2	ruby system pattern holding multiple types objects class building accounting system rails account account will postings account account journal object describe entire main method account will call transfer method journal generate details system twist want allow account pledge points account pledging account will longer access points later account authorizes transaction account think learning accounts user settled pledged when pledges points transferred settled account pledged account when approves points will transferred pledged account settled implement column account enum pledged look right when user wants pledge find account type send point type account send change account cant transfer decision behalf user multiple accounts user transfer functionality user class will select send points account want tight couple financial system general part system wrapper class relationship user chose account send point seems unnecessarily method classes transfer settle will lookup appropriate account types transfer look right think revise posting handle smoothly advice	ruby-on-rails;ruby;design-patterns	1	enable	https://stackoverflow.com/questions/7378753/ruby-design-pattern-holding-multiple-types-of-objects-in-one-class	Interface design pattern (User Interface, Frameworks)	1	yes	program related	long term	yes	no
3	pattern swing exchange design design frame example frame frame opens frame will text edit pressing button frame closed want text variable frame want font chooser option frame shall html editor frame click table will open frame html editor using purpose when close frame pressing button want html text string frame content accordingly frame modal dialog	java;swing;design-patterns;iframe	1	enable	https://stackoverflow.com/questions/9217611/the-mvc-pattern-and-swing	Interface design pattern (User Interface, Frameworks)	7	yes	theoretical	long term	yes	yes
4	pattern radio buttons selecting controls noticed common scenario development client group radio buttons clicking radio button means want certain controls take lets text illustrate radio buttons underscores represent text boxes radio button select text boxes translated requirements user clicks values text wiped whenever user starts typing text radio button controlling automatically selected implies wiping text values problem programming correctly surprisingly hard to solve handles cases events fire pattern followed view show data model facing problem view showing data model data data model birth date view lets user enter birth date date goes model right text textbox once first user types birth date cannot parse sort valid date controller react turn events problem view model disconnect model contains view view supposed represent model instead showing text stored property reconcile disconnect view showing model contains view view showing actually needs need store value somewhere cant save setting text property cant stored model user made date view model even complicated when view nothing data model date days controller parse string convert date controller stores date model discrepancy view model view model user enter dates cannot date mercury idea turn date really going just going tell user enter dates normal person question show user text view user controller display model case model does contain mercury house cannot model birth date cannot hold when user tries save check birthdate valid going correctly look 5 model assume user entered proper date except	design-patterns;user-interface;client-side	1	enable	https://stackoverflow.com/questions/13883054/how-can-i-implement-parts-of-views-that-are-conditional-on-model-properties/1388924#1388924	Interface design pattern (User Interface, Frameworks)	1	yes	program related	long term	yes	yes
5	create left panel right panel load panels panels button when pressed load right panel press panel change load schema save state when press loads panel controls example write name next press loads panel controls finally return panel next user panel contains name empty vice versa controls panel something change panel share information load panel list users user change panel show label users want reuse list count want execute window controller associate window structure when panel loads check controller exists create instance save next time user panel controller load state last time works problem exchange information panels controllers know pass controllers prefer want spring 6 framework know design store different types database building want users create albums stuff albums stuff album mainly images videos tracks every piece item album table content table content table rows pieces content want store thought creating table foreign content table turn foreign album table content table information type content stored image video text video table info piece content example image following table question store pieces content database right solution scalable albums users will easy content types future means expert developer really want learn things properly real life examples across different solution images stored media table store video example create different table different content types want store tables images videos songs create media images videos table content table saving texts pieces html application post content type album item song image really help giving advice 7 tackle dilemma enterprise reliable queue working think need queue already intend enterprise makes sense queue useful commands queue producers insert records consumers process remove records easily producers consumers scale horizontally scale bottleneck will queue stored single shard good distribute queue across multiple shards solution multiple queues somehow ensure queue hashes different shard require producers consumers queue keys changed when ideal intending queue batch insert records database producers servers receiving requests generating records during peak traffic times database able keep rate single inserts cant just buffer requests servers buffered records lost when server fails queue provides fault tolerance replication consumers multiple records queue performing insert lessen load database wont scale enough better solution missing good advice greatly starting creating library using method discussed available 8	model-view-controller;language-agnostic;design-patterns	1	enable	https://stackoverflow.com/questions/11586-do-you-use-design-patterns	Interface design pattern (User Interface, Frameworks)	15	yes	theoretical	long term	yes	yes
6	design store different types database building want users create albums stuff albums stuff album mainly images videos tracks every piece item album table content table content table rows pieces content want store thought creating table foreign content table turn foreign album table content table information type content stored image video text video table info piece content example image following table question store pieces content database right solution scalable albums users will easy content types future means expert developer really want learn things properly real life examples across different solution images stored media table store video example create different table different content types want store tables images videos songs create media images videos table content table saving texts pieces html application post content type album item song image really help giving advice 7 tackle dilemma enterprise reliable queue working think need queue already intend enterprise makes sense queue useful commands queue producers insert records consumers process remove records easily producers consumers scale horizontally scale bottleneck will queue stored single shard good distribute queue across multiple shards solution multiple queues somehow ensure queue hashes different shard require producers consumers queue keys changed when ideal intending queue batch insert records database producers servers receiving requests generating records during peak traffic times database able keep rate single inserts cant just buffer requests servers buffered records lost when server fails queue provides fault tolerance replication consumers multiple records queue performing insert lessen load database wont scale enough better solution missing good advice greatly starting creating library using method discussed available 8	dependency-injection;state;javafx-8	1	enable	https://stackoverflow.com/questions/2607336/create-a-javafx-application-with-subpanels	Interface design pattern (User Interface, Frameworks)	1	yes	theoretical	long term	yes	yes
7	design store different types database building want users create albums stuff albums stuff album mainly images videos tracks every piece item album table content table content table rows pieces content want store thought creating table foreign content table turn foreign album table content table information type content stored image video text video table info piece content example image following table question store pieces content database right solution scalable albums users will easy content types future means expert developer really want learn things properly real life examples across different solution images stored media table store video example create different table different content types want store tables images videos songs create media images videos table content table saving texts pieces html application post content type album item song image really help giving advice 7 tackle dilemma enterprise reliable queue working think need queue already intend enterprise makes sense queue useful commands queue producers insert records consumers process remove records easily producers consumers scale horizontally scale bottleneck will queue stored single shard good distribute queue across multiple shards solution multiple queues somehow ensure queue hashes different shard require producers consumers queue keys changed when ideal intending queue batch insert records database producers servers receiving requests generating records during peak traffic times database able keep rate single inserts cant just buffer requests servers buffered records lost when server fails queue provides fault tolerance replication consumers multiple records queue performing insert lessen load database wont scale enough better solution missing good advice greatly starting creating library using method discussed available 8	php;laravel;database-schema;software-design	2	full, produce, plan, combine	https://stackoverflow.com/questions/3409090/sofw-are-design-how-to-store-different-types-of-content-in-a-database	API/Schema-Design-Pattern Data Access Design	1	no	program related	long term	yes	yes
8	enterprise reliable queue working think need queue already intend enterprise makes sense queue useful commands queue producers insert records consumers process remove records easily producers consumers scale horizontally scale bottleneck will queue stored single shard good distribute queue across multiple shards solution multiple queues somehow ensure queue hashes different shard require producers consumers queue keys changed when ideal intending queue batch insert records database producers servers receiving requests generating records during peak traffic times database able keep rate single inserts cant just buffer requests servers buffered records lost when server fails queue provides fault tolerance replication consumers multiple records queue performing insert lessen load database wont scale enough better solution missing good advice greatly starting creating library using method discussed available 8	architecture;redis;queue;software-design	2	full, produce, plan, combine	https://stackoverflow.com/questions/45262173/redis-enterprise-sharding-reliable-queue	API/Schema-Design-Pattern Data Access Design	1	yes	theoretical	temporary	yes	yes
9	async image upload pattern building site user uploads image part process listing item want show preview image upload form aesthetically pleasing will help users uploading describing order show preview image need upload server sandboxing browsers cannot image data except submitting form however store images using identifiers based data user inputs order keep organized case sanitize name file name folder change naming scheme files order remove dependency user input file uploading resizing image expensive operation want resources unnecessarily upload image resizing generating thumbnails form containing information item submitted application will stream data resize generate thumbnails resupload images seems somewhat feel pretty common problem wonder anyone patterns advice handle image uploads regard 9	design-patterns;amazon-s3;flask;image-uploading	2	full, produce, plan, combine	https://stackoverflow.com/questions/11379867/async-image-upload-pattern	API/Schema-Design-Pattern Data Access Design	2	yes	theoretical	long term	yes	yes
10	factory design pattern state design pattern user enter link image video according type media image video will determine display thumbnail entered media link example link entered video generate thumbnail video generate html code snippet will display image reduce size return html snippet smaller image link generate embed html type link will probably method will return thumbnail image will return html code image avoid using statements able grow types media links decided state capture link whatever media type within object state time either methods compatible image design patterns best practice image processing real time application first image processing image less application three threads addition main thread running image acquisition second image processing third robot threads image queue share camera enqueues images robot dequeues processed images imaging processor dequeues images enqueues processed images restriction noticed processed images sequence meaning keeping order images image design pattern best practice applied architecture 10	if-statement;design-patterns;factory-pattern;state-pattern	1	enable	https://stackoverflow.com/questions/24014653/remote-functionality-provided-by-spring-security	API/Schema-Design-Pattern Data Access Design	1	yes	program related	long term	yes	yes
11	factory design pattern state design pattern user enter link image video according type media image video will determine display thumbnail entered media link example link entered video generate thumbnail video generate html code snippet will display image reduce size return html snippet smaller image link generate embed html type link will probably method will return thumbnail image will return html code image avoid using statements able grow types media links decided state capture link whatever media type within object state time either methods compatible image design patterns best practice image processing real time application first image processing image less application three threads addition main thread running image acquisition second image processing third robot threads image queue share camera enqueues images robot dequeues processed images imaging processor dequeues images enqueues processed images restriction noticed processed images sequence meaning keeping order images image design pattern best practice applied architecture 10	+++ design-patterns	2	full, produce, plan, combine	https://stackoverflow.com/questions/12019140/design-patterns-or-best-practice-for-image-acquisition-and-image-processing-in	API/Schema-Design-Pattern Data Access Design	2	yes	theoretical	long term	yes	yes
12	best practice objects design code design doing refactoring website cosplay site upload images self costumes original site refactoring done site switch left column list site images show name different characters images anime click anime will list characters within given anime images show character name click galleries images click character name will list galleries given character given anime information gallery image count click gallery list images database look design imagine different create site first site need select table result need count select characters galleries plan create will plan lazy load cant decide best solution better doing priority good performance minimum lose features code using service pattern design absolute willing change increase performance hope describe question good enough understand mean away anything missed 12	asp.net-mvc-2;responsibility-pattern;design-patterns	2	full, produce, plan, combine	https://stackoverflow.com/questions/1690965/best-practice-for-asp-net-web-app-design	API/Schema-Design-Pattern Data Access Design	1	yes	theoretical	long term	yes	yes
13	chain scope chain responsibility general common used pattern handlers itself handler passing functionality action example scenarios violate chain responsibility pattern does second scenario suits chain responsibility violates 13 example handlers itself responsible actions pass information themselves handler mechanism suits chain responsibility	design-patterns;netty;handlers;chain-of-responsibility	3	valid, citation	https://stackoverflow.com/questions/45979004/chain-of-responsibility	Object Oriented Design Principles (SOLID, Chain of Responsibility)	1	yes	theoretical	long term	yes	yes

SONO	Multiple	Tags	Dominant Topic	Topic Terms	URL	Label	No. of Answers (Number)	Is Best Answer Selected? (Yes/No)	Is the question Theoretical or Program related? (Theoretical/Programming)	Is the solution temporary or long term? (Temporary/Long Term)	Is the discussion related to design? (Yes/No)	Is label representative? (Yes/No)	
14	<p>many rather distinct contexts closed want improve update question answered facts citations editing post closed years doing research concepts having issue trying understand exactly gone numerous stack overflow posts topic really able pinpoint satisfying seen discussions differences encapsulation naturally started started thinking ion terms having particular class works providing abstraction class posts steered read posts related answers portraying abstraction inheritance context specifically using interfaces abstract classes provide abstraction certain entity class assumed abstraction given manner allow developers extend objects appropriately according guidelines outlined abstraction posts lead sure just completely missing point getting pretty confusing every answer seems slight variation definitely contexts crucial object oriented programming really wanted definition abstraction brings point does abstraction work multiple contexts does abstraction portray away unnecessary details done interfaces implementation class exposing class links included above contains following quote problem precise definitions concepts words themselves multiple meanings even within context object orientation just abstraction abstract even definition abstract thanks guidance rather really aware primarily opinion found flag entails question less valid slow questions regarding abstraction think considered less actually pinpointing different contexts think abstraction makes sense seen questions just abstraction think even broader question</p>			switch, break, handler, chain, validation, future, determine, condition, refactor, refactor, validate, concern, violate, responsibility, choose, encapsulate, character, prototype, valid, citation	https://stackoverflow.com/questions/2645627/difference-between-oop-basics-vs-solid	Object Oriented Design Principles (SOLID, Chain of Responsibility)	2	yes	theoretical	long term	yes	yes	
15	<p>moving data interfaces violating solid principles best move data interfaces violating solid principles overthinking intend dogmatic regard solid principles wanted input refactoring shopping cart solid method wrote seems code smell maybe class looks something simplified part seems method something seems funky cant think alternative better properties added need passed view sense interface segregation thought adding methods seems violation single responsibility principle sense accepts conversion seems unnecessary overhead simple solutions define explicit cast lets change method cleaner problem approach violates single responsibility principle coupling class considered extension method instead cast conversion better just concrete class implement interface properties maybe just rewrite remove options seem violations single responsibility principle maybe solution will become obvious guess question boils down best move data interfaces without violating solid principles suggestions maybe just move worry dogmatic solid answered question maybe belongs hope subjective case helpful interfaces look added post attributes clarify</p>	oop:inheritance; design-patterns; abstraction		switch, break, handler, chain, validation, future, determine, condition, refactor, refactor, validate, concern, violate, responsibility, choose, encapsulate, character, prototype, valid, citation	https://stackoverflow.com/questions/1773436/moving-data-interfaces-without-violating-solid-principles	Object Oriented Design Principles (SOLID, Chain of Responsibility)	3	no	program related	long term	yes	yes	
16	<p>handler handle request chain pattern going chain responsibility pattern definitions however want know series handlers chain request handler handlers last handlers will process request beginning handlers failed against pattern will explain example assume online application allows users search books when user submits request system does search shows results assume functionality implemented pattern first handler takes request parameters book category adds additional parameters users picked request does certain second handler takes output first handler took category create list results results transferred third handler third handler will order results filter change language preceding handler failed following handlers will usage against concepts good pattern</p>	java:design-patterns		switch, break, handler, chain, validation, future, determine, condition, refactor, refactor, validate, concern, violate, responsibility, choose, encapsulate, character, prototype, valid, citation	https://stackoverflow.com/questions/1654721/don-t-internal-sealed-classes-互相-perpendicular-to-proper-di	Object Oriented Design Principles (SOLID, Chain of Responsibility)	1	no	theoretical	long term	yes	yes	
17	<p>better wrap chain functionality directly class focusing learning programming principles patterns chain examples found seem contradict placement directly class will just seems really really idea trying learn previously patterns find lots unrealistically simple examples first thought case examples found lots googling find examples coming close imagined started checking articles find even short sentence handler functionality separate class using thinking maybe deal chain responsibility concrete class just again just seems really really wrong want class chain responsibility pattern somewhere else chain responsibility functionality built directly going pages pages google articles chain responsibility finding nothing topic eventually find example implements functionality wrapper class pretty delegates concrete class wrapper class concrete class used needing chain responsibility functionality question will real world implementations chain responsibility principle maybe just directly class maybe just misunderstand situations pattern actually used really able compare understand method decided write examples using methods below pasted important parts linked full</p>	java:design-patterns; software-design; design-principles		switch, break, handler, chain, validation, future, determine, condition, refactor, refactor, validate, concern, violate, responsibility, choose, encapsulate, character, prototype, valid, citation	https://stackoverflow.com/questions/7212072/use-closures-to-make-methods-private-in-prototype/7213015#7213015	Object Oriented Design Principles (SOLID, Chain of Responsibility)	3	yes	program related	long term	yes	yes	
18	<p>design pattern design using closed want improve update answered facts citations editing post closed years having trouble designing module anybody help will hard maintain kind module think test skill design pattern basically agricultural project application need design module calculation takes different crops involved maize tomato okra crops different trait measurement scale lies integer lets planted crop done measurement noted down traits want sort measurement measurements simple example crop maize recorded observations traits examples actual name recorded observations logs system selects crops enters data observations calculate either average data entered centre simple complexity comes when different formulas trait formula based calculate value depend traits different crop different able whenever user selects crop will check specific traits calculations special trait either average based entry hard suggestions better handling code needs handle simple complex calculations easy average value entered problem comes when complex calculations crop different traits formulas calculate check crop complex trait hardcode trait name complex tell design using java oops different crops calculations specific crops will code below class crop think application supports crops user logs system link above exercise better crop traits crop know distributed updating object objects structure current program organized object list mention objects mention object constraints list word objects words identified position docs text store information text processing program user interaction word objects inside mention accessed modified value example update sense user interaction mention problem several mentions belong share words words when word updated update corresponding word contained mentions words words exact location text updated together stored separately mentions update change others approach used when word inside mention modified retrieve mentions stored reference update corresponding word mention contains requires loop equals checks update quite processing second approach think store separate word lists mentions single list words store mention indices words belong mention store list when updating word will call update function docs reference update docs list however problem lies function returns whole list words mention return list words using indices pick actual words inside docs list needed words inside mention modified mentions just shortly alternatively check word updates loop update requires loop words mention seems weird time retrieving list long wait better solution update problem help appreciated necessary will part code</p>	java:oop:design-patterns		switch, break, handler, chain, validation, future, determine, condition, refactor, refactor, validate, concern, violate, responsibility, choose, encapsulate, character, prototype, valid, citation	https://stackoverflow.com/questions/2116290/which-design-pattern-and-how-to-design-using-oop-this-scenario	Object Oriented Design Principles (SOLID, Chain of Responsibility)	3	no	theoretical	long term	yes	yes	
19	<p>adding namespace developing parts decoder takes string builds proper object programmer objects standard will application standard application objects will prior development will live namespace name different strings world produce object properly hello world value decoder will check standard prototypes check application namespaces assume will just question applies multiple namespaces first problem find namespaces lets find options creating method decoder stores namespaces check later method called every time namespaces created however introduces coupling scripts define namespaces require decoder already available want option having variable namespaces register themselves upon construction decoder browses option similar first pollutes namespace want third option namespaces decoder when loading browse window store locally namespaces label mean adding property namespaces example value case worst used simply loop window keys check store want uninteresting list option making namespaces instance super namespace imagine object will namespace even inherit super namespace necessary lets extend fourth option object part standard will call super lets standard objects following code including namespaces objects application using standard objects giving custom business execute following code example check working will think options answer fourth option extra ways important code pure enforcing type safety inherited members inherited classes trying adhere good design principles design patterns developing mine find multiple design architecture issues always want find implementing canonical hopes building highly maintainable flexible become better suppose classes character weapon derived classes sword staff weapon warrior mage character hypothetical related actual every character weapon every implementation character know implementation weapon will instance know want enforce runtime every instance warrior will weapon type sword course every time want weapon object properly inside warrior perform cast believe great practice means prevent myself missing type safety ideal solution able override weapon property warrior class sword weapon property type safety user uses warrior character sword weapon property type safety seem kind classical problem name solution case know name problem solutions links good reading material helpful kind class design propose order maintain functionality enforce type safety elegant idiomatic reading</p>	cf:list:design-patterns		window, proper, sample, import, word, wrap, play, wrapper, script, tutorial, empty, folder, hope, namespace, render, properly, advice, actual, guess, circular	https://stackoverflow.com/questions/1848231/distributing-updating-of-object-collections	Class Design Patterns (Inheritance, Polymorphism)	4	advice, actual, guess, circular	program related	long term	yes	yes	
20	<p>adding namespace developing parts decoder takes string builds proper object programmer objects standard will application standard application objects will prior development will live namespace name different strings world produce object properly hello world value decoder will check standard prototypes check application namespaces assume will just question applies multiple namespaces first problem find namespaces lets find options creating method decoder stores namespaces check later method called every time namespaces created however introduces coupling scripts define namespaces require decoder already available want option having variable namespaces register themselves upon construction decoder browses option similar first pollutes namespace want third option namespaces decoder when loading browse window store locally namespaces label mean adding property namespaces example value case worst used simply loop window keys check store want uninteresting list option making namespaces instance super namespace imagine object will namespace even inherit super namespace necessary lets extend fourth option object part standard will call super lets standard objects following code including namespaces objects application using standard objects giving custom business execute following code example check working will think options answer fourth option extra ways important code pure enforcing type safety inherited members inherited classes trying adhere good design principles design patterns developing mine find multiple design architecture issues always want find implementing canonical hopes building highly maintainable flexible become better suppose classes character weapon derived classes sword staff weapon warrior mage character hypothetical related actual every character weapon every implementation character know implementation weapon will instance know want enforce runtime every instance warrior will weapon type sword course every time want weapon object properly inside warrior perform cast believe great practice means prevent myself missing type safety ideal solution able override weapon property warrior class sword weapon property type safety user uses warrior character sword weapon property type safety seem kind classical problem name solution case know name problem solutions links good reading material helpful kind class design propose order maintain functionality enforce type safety elegant idiomatic reading</p>	javascript:design-patterns;prototype		window, proper, sample, import, word, wrap, play, wrapper, script, tutorial, empty, folder, hope, namespace, render, properly, advice, actual, guess, circular	https://stackoverflow.com/questions/1389372/adding-a-qualifier-label-to-a-namespace	Class Design Patterns (Inheritance, Polymorphism)	4	advice, actual, guess, circular	program related	long term	yes	yes	
21	<p>design pattern design using closed want improve update answered facts citations editing post closed years having trouble designing module anybody help will hard maintain kind module think test skill design pattern basically agricultural project application need design module calculation takes different crops involved maize tomato okra crops different trait measurement scale lies integer lets planted crop done measurement noted down traits want sort measurement measurements simple example crop maize recorded observations traits examples actual name recorded observations logs system selects crops enters data observations calculate either average data entered centre simple complexity comes when different formulas trait formula based calculate value depend traits different crop different able whenever user selects crop will check specific traits calculations special trait either average based entry hard suggestions better handling code needs handle simple complex calculations easy average value entered problem comes when complex calculations crop different traits formulas calculate check crop complex trait hardcode trait name complex tell design using java oops different crops calculations specific crops will code below class crop think application supports crops user logs system link above exercise better crop traits crop know helper traits companion objects first describe thinking functor definition example defined example small changes less functor module definition want define module exact types instance list plus want define generic function accepts functor implicitly better define close possible functor trait think best place consideration companion object definition functor implicitly client code whos want module functions want enrich existing types able functions part existing types achieve implicit final class allows convert generic type name functions differently avoid using mistake function exists different want define implicit convertor first option comes mind companion object looking code found mostly traits traits compose import word example trait extend client code import easier import companion objects traits application supports crops user logs system link above exercise better crop traits crop know</p>	oop:inheritance; design-patterns; polymorphism		window, proper, sample, import, word, wrap, play, wrapper, script, tutorial, empty, folder, hope, namespace, render, properly, advice, actual, guess, circular	https://stackoverflow.com/questions/29521381/enforcing-type-safety-of-inherited-members-in-inherited-classes	Class Design Patterns (Inheritance, Polymorphism)	4	advice, actual, guess, circular	program related	long term	yes	yes	Sparked interesting thoughts related to design principles
22	<p>design pattern design using closed want improve update answered facts citations editing post closed years having trouble designing module anybody help will hard maintain kind module think test skill design pattern basically agricultural project application need design module calculation takes different crops involved maize tomato okra crops different trait measurement scale lies integer lets planted crop done measurement noted down traits want sort measurement measurements simple example crop maize recorded observations traits examples actual name recorded observations logs system selects crops enters data observations calculate either average data entered centre simple complexity comes when different formulas trait formula based calculate value depend traits different crop different able whenever user selects crop will check specific traits calculations special trait either average based entry hard suggestions better handling code needs handle simple complex calculations easy average value entered problem comes when complex calculations crop different traits formulas calculate check crop complex trait hardcode trait name complex tell design using java oops different crops calculations specific crops will code below class crop think application supports crops user logs system link above exercise better crop traits crop know helper traits companion objects first describe thinking functor definition example defined example small changes less functor module definition want define module exact types instance list plus want define generic function accepts functor implicitly better define close possible functor trait think best place consideration companion object definition functor implicitly client code whos want module functions want enrich existing types able functions part existing types achieve implicit final class allows convert generic type name functions differently avoid using mistake function exists different want define implicit convertor first option comes mind companion object looking code found mostly traits traits compose import word example trait extend client code import easier import companion objects traits application supports crops user logs system link above exercise better crop traits crop know</p>	java:oop:design-patterns		window, proper, sample, import, word, wrap, play, wrapper, script, tutorial, empty, folder, hope, namespace, render, properly, advice, actual, guess, circular	https://stackoverflow.com/questions/2116290/which-design-pattern-and-how-to-design-using-oop-this-scenario	Class Design Patterns (Inheritance, Polymorphism)	4	advice, actual, guess, circular	theoretical	long term	yes	yes	
23	<p>companion objects find using traits flexible first cause compose perhaps miss something wrong questions</p>	scala:design-patterns:implicit; conversion:scalaz		window, proper, sample, import, word, wrap, play, wrapper, script, tutorial, empty, folder, hope, namespace, render, properly, advice, actual, guess, circular	https://stackoverflow.com/questions/50791967/where-to-put-implicit-and-helper-functions-traits-vs-companion-objects	Class Design Patterns (Inheritance, Polymorphism)	4	advice, actual, guess, circular	program related	long term	yes	yes	

S N O	Text	Tags	Dominant Topic	Topic Terms	URL	Label	No. of Answers (Number)	Is Best Answer Selected? (Yes/No)	Is the question Theoretical or Program related? (Theoretical/Programming)	Is the solution temporary or long term? (Temporary/Long Term)	Is the discussion related to design? (Yes/No)	Is label representative? (Yes/No)
36	sign function cant understand sign function builtin consider signs sister python even function sign bother write when just write sign directly latter clear sign whereas remember sign sign does provide anything readable greatly readable language python plus python designer builtin sign when need just even better nonnumerical stuff just course required sorted ascending boolean instead integer comparator clear positive when whereas remember convention positive when first bigger around course makes sense reasons when sorting nonnumerical things want sort stable possible using simply question python designers decide leave sign function language heck bother parent missing peter comment fair enough python years python needed countless times last straw broke camels backs around times needed pass idiom lambda worked sign just hope agree sign useful even bought view bother defining instead sign useful sign	python;language-design		comment, developer, benefit, true, people, engine, plugin, operator, machine, interested, syntax, experience, signature, hear, recently, draw, short, important,	https://stackoverflow.com/questions/1986152/why-doesnt-python-have-a-sign-function	Language Design		13 yes	theoretical	long term	yes	
37	search efficiency partial matches building decoder binary file need search binary file specifically partial byte buffer probably going choose frame headers means using efficient string search algorithm decided problem headers partial bytes bits approaches store children interval lines rather hashes given example pattern first node will store range node will store range unfortunately greatly expands number suffix links dictionary suffix links among nodes think solution equivalent following partial bytes pattern possible matching byte patterns obviously increasing number possible patterns greatly increases search bytes bits using module individual bits backed bytes individually accessible unfortunately increases search space fold search pattern pattern multiple bytes manually check three remaining bits using standard integer shifting greatly increases number pattern occurrences search crazy think approaches exactly complexity approach complexity length search text total length search patterns number matches detects overlapping patterns necessary cant happen patterns overlap	python;algorithm; design-patterns; time-complexity; string-search		match, performance, term, column, loop, adapter, entry, track, long, dispose, repeat, reduce, amount, huge, split, efficient, grow, increase, key, active	https://stackoverflow.com/questions/45741658/multi-le-string-search-efficiency-for-partial-matches	Efficient Algorithm Data Access Design (Strings, Lists, Data Structures)		0 no	theoretical	long term	yes	
38	comparing objects efficiently trying compare fact data fact made predicate list argument class made variable subclasses examples numeric arguments length length length examples string arguments length length value matches exactly argument matches arguments variable construct generic argument type setting appropriate symbol onto matching examples good fall colour black length fall colour black colour brown size matches example soft colour brown size soft colour brown missing arguments define object predicate implemented methods match argument object struggling implement good method compare fact different experimented using instead arguments tried adding field field instead including within argument grateful advice ways overhaul data structure matching super slow think timeouts multiple clients java server multiple clients period clients sends alive packet server moment server alive packets period change device device runtime server clients want generate alert when clients send alive packet aliveness used parts application quicker notice better ideas select task checks every clients last alive packet timestamps current time generate alert alerts call method period smaller minimum actually seems better however unnecessarily check clients alive clients period change minute task every minute least check clients above minute period redundant minimum client periods decrease decrease tasks period task clients check last alive packet timestamps current time sleep clients period time clients number goes high will dozens task will sleep time doubt elegant idiom pattern kind situation think watchdog kind implementation suite however something java	java;list;oop;set; software-design		match, performance, term, column, loop, adapter, entry, track, long, dispose, repeat, reduce, amount, huge, split, efficient, grow, increase, key, active	https://stackoverflow.com/questions/4454851/loom-args-a-list-set-of-objects-efficiently	Efficient Algorithm Data Access Design (Strings, Lists, Data Structures)		1 yes	program related	long term	yes	
39	designed performance mind advisable design architecture performance mind mean ready used designing components just follow good principles just ensure component extendable tweak design when performance issues although land performance issues tweaking little design although complex makes performance issues cakewalk need tweak software tweaking often simple design above things listed above trying tune performance software running performance issues rephrase question design software answer saying depends environment software expected reason clients software just seem want time plan software constantly running environments redesign software when sense question troubling weak answer	java;architecture; software-design		match, performance, term, column, loop, adapter, entry, track, long, dispose, repeat, reduce, amount, huge, split, efficient, grow, increase, key, active	https://stackoverflow.com/questions/48029344/timeouts-for-multiple-clients-in-java	Efficient Algorithm Data Access Design (Strings, Lists, Data Structures)		3 no	theoretical	long term	yes	
40	accelerate process divide list sublists based pattern sublists lists sparse matrix sparse sparse sparse sparse want list sublist sublists made list sparse matrix length parallel later sublists values input lists sections following gives list chunks filter order sublist different index value works lots input millions entries takes need program less hours needs calculate times does python command exist lists depending pattern first sublist thank view matrix	language-agnostic; oop;design-principles		match, performance, term, column, loop, adapter, entry, track, long, dispose, repeat, reduce, amount, huge, split, efficient, grow, increase, key, active	https://stackoverflow.com/questions/1548026/about-software-be-designed-with-performance-in-mind	Efficient Algorithm Data Access Design (Strings, Lists, Data Structures)		10 no	theoretical	long term	yes	
41	popular today week design pattern system displays entries ordered three fields popular today week time entry viewed score incremented thus changing entry viewed times today scores will current month simply fields associated entry today week month time entry viewed three scores incremented scores reset current week week current calendar month month score works uses little space ideal reasons current period value reset once meaning every ranking reset daily scores true week month month scores losing existing ranking data month usually falls inside week monthly scores reset during week leading weekly scores higher monthly rolling hourly counts every hour month used calculate scores current based current hour view placed stats calculator today last values week score last values finally month last major issue solution requirements gone using values current solution using values even change going using memory solution memory usage entry jumped anyone suggest solution meet resolve issues current solution using thanks	python;list;design-patterns;sparse-matrix		match, performance, term, column, loop, adapter, entry, track, long, dispose, repeat, reduce, amount, huge, split, efficient, grow, increase, key, active	https://stackoverflow.com/questions/45869453/how-to-accelerate-a-process-to-divide-a-list-of-sublists-based-on-a-pattern-in	Efficient Algorithm Data Access Design (Strings, Lists, Data Structures)		1 yes	program related	long term	yes	
42	building progressive main shell rest will supply data needed will handle rest google seems problematic browser needs accounted want user login persist closing down done research possible ways going however ensure going wrong based authentication part sends username password receives http cookie session session needs stored database issue option cookies automatically sent browser therefore need protection place using synchronizer token pattern token generated every time state changing request made post means application needs supply token every request send determined ideal user send multiple post requests quick succession making fail resulting user method limiting policy domain adding header requirement technically stop however unsure secure token based authentication user sends username password token issued needs stored cookie using means vulnerable token stolen attacker impersonate user completely using cookies will issue resolve considered double submit cookie method refresh every time reassess creates window attacker find clear method best based authentication token authentication user sends username password session created http cookie browser sent back user authenticate requests whatever happens expires calls again acquire endpoint need protected however impact usability seems large amount articles claiming insecure use high organisations amazon recommending users store token	authentication; session; progressive-web-apps;api-design; jwt-auth		response, back, retrieve, complete, http, endpoint, token, fetch, status, connect, unique, transaction, storage, cache, network, header, wait, limit, party, mechanism	https://stackoverflow.com/questions/6001390/api-authentication-for-owa	Authentication-Design User Interaction Design		0 no	theoretical	long term	yes	
43	angular http interceptors auth tokens trying write http interceptor will attach header order authorize auth service requests stores token future configuring pushing interceptor onto interceptor depends auth service order hold token send time turn auth service depends http order send initial request authenticate retrieve auth token circular dependency graph looks something depends http depends elegant around thought using intermediate service ultimately just extending dependency graph something needs fundamentally possible something resolve reconfigure http auth token retrieved	javascript; angularjs; authentication; dependency-injection		response, back, retrieve, complete, http, endpoint, token, fetch, status, connect, unique, transaction, storage, cache, network, header, wait, limit, party, mechanism	https://stackoverflow.com/questions/28973199/angular-auth-interceptors-for-auth-tokens	Authentication-Design User Interaction Design		1 yes	theoretical	long term	yes	
44	proper client restful endpoint updating multiple entities single batch standard rest request update single entity example document using endpoint looks something http status codes communicate client using http status successful update http document specified found http problem updating record issue restful potentially extremely high usage performance create endpoint will accept multiple document entities update single input user submits documents able able successfully update remaining failing issue commit successfully updated documents communicate user updates succeeded ones approach submitfully update return http response object return client include list documents succeeded list documents failed failed include reason along maybe http status code failed returning http requests failed approach counts client inspect list failed documents problems fear user will http assume everything option client able successfully update fails return http status code failed example failed specified found return http failed unavailable return http approach problems example documents fail different reasons http status code returned does sense return example http status request successfully updated rest guidelines give	rest;asp;net-web-api;restful-architecture;api-design		response, back, retrieve, complete, http, endpoint, token, fetch, status, connect, unique, transaction, storage, cache, network, header, wait, limit, party, mechanism	https://stackoverflow.com/questions/2469898/propo-ressponse-to-client-for-restful-pul-endpoint-for-updating-multiple-entit	Authentication-Design User Interaction Design		1 yes	program related	long term	yes	
45	suggestions issue batch updates recommended approaches issue	java;design-patterns;timer; observer-pattern; timer-task		response, back, retrieve, complete, http, endpoint, token, fetch, status, connect, unique, transaction, storage, cache, network, header, wait, limit, party, mechanism	https://stackoverflow.com/questions/1567398/java-combine-observer-pattern-with-timer-task	Authentication-Design User Interaction Design		2 yes	theoretical	short term	yes	no
46	java combine observer pattern timer task previous post used observer pattern class flight status time time late flights class observer observer object method will update status flight status flights thinking using timer tasks change status flights chosen times updated status flights want able flight status displays screen just changing timer task sure doing correctly will concurrency will problem flights whose status will change batches batch size flight flights change states later couple seconds later change status batch flights remain related post restricted object types added using composite pattern building custom framework implemented composite pattern build object tree representing page rendered block object inherit class working great rendering snap help little need validate classes blocks added blocks example distinct block types consider keep mind blocks inherit single block classifications behavior concrete classes contained additional generic block type added final child blocks added parent block specific type child added added generic child added specific parent block share properties generic parent goal given concrete block class name generate list concrete block classes added solutions proposed property called block abstract defines generic final parent child code logic validate function gets called solution number reasons mainly give clear path defining concrete parents children allowed instance concrete class parent blocks added added additional properties added build relationships properties examples derive block classification making approach impractical property block defines list concrete classes added ease values specific meaning represents block types null value means blocks added cases some delimited string class names provided classes classes inherit leaning towards direction gives operator checks when concrete classes extend base check will return results parent class hesitation method definitely solves problem given concrete block class name generate list fee limits future search down tree back given child blocks generate list parent blocks added sure ever problem question masses best wondering getting list classes first place framework implements scanning directory tree free feeds class names dynamically instantiating	php;oop;design-patterns		response, back, retrieve, complete, http, endpoint, token, fetch, status, connect, unique, transaction, storage, cache, network, header, wait, limit, party, mechanism	https://stackoverflow.com/questions/23831731/restricting-object-types-that-can-be-added-to-each-other-using-the-composite-pat	Authentication-Design User Interaction Design		1 yes	program related	long term	yes	no

SO	Question	Tags	Dominant Topic	Topic Terms	URL	Label	No. of Answers (Number)	Is Best Answer Selected? (Yes/No)	Is the question Theoretical or Program related? (Theoretical/Programming)	Is the solution temporary or long term? (Temporary/Long Term)	Is the discussion related to design? (Yes/No)	Is label representative? (Yes/No)
	memento design pattern save state multiple objects memento supposed understand memento state memento pattern used store different previous states restore object previous states lets multiple objects comes attributes stay throughout life individual object change need object save previous states back apply memento pattern idea memento pattern classes memento create state caretaker stores previous states object mementos originator creates mementos gets states mean object instance previous caretaker instance list previous states caretaker mementos previous states objects attributes current state previous ones objects caretaker originator instance originator used create mementos implemented misunderstanding look something memento making memento class originator class hold attributes instead encapsulating attributes inside class object instance hold instance originator instead though originator contain current state attributes object instance know correct implementing examples online mementos store state-of-just string	design-patterns; memento		dynamic, stuff, dynamically, delete, stack, email, general, concept, purpose, helper, describe, head, hard, info, behaviour, world, actual, thought, locator, student	https://stackoverflow.com/questions/40795754/how-to-use-the-memento-design-pattern-to-save-the-state-of-multiple-objects	Separation of Concerns Design	1	no	theoretical	long term	yes	
59	algorithm migration linear problem given number schools school holds number teachers according needs scholar year teachers changing positions schools currently teaching according ordered list change school possible school possible school remembering school exactly number teachers needs less teacher importance number unique teachers school time having higher importance number will desired school unable migrate teachers following list keep initial school answering sorry demand affordable year afford migration afford mean change position teacher best best according list desired school algebraic modeling positive integers function function function used defined function available defined function entity ordered nonempty list locations destinations willing move bijection importance entity unique importance number asked task find function positioning affords following belongs pell possible solutions keep solution matches destinations order case exclude possible solutions keep solution matches importances order case exclude none desired destinations possible	algorithm;design-patterns;algebra; bipartite		dynamic, stuff, dynamically, delete, stack, email, general, concept, hard, info, behaviour, world, actual, thought, locator, student	https://stackoverflow.com/questions/29344535/algorithm-for-conditioned-migration	Separation of Concerns Design	1	no	theoretical	long term	yes	
60	difficult architectural problem involving recurring payments future events looking guidance architect elegant become thorny problem although using ruby rails think problem largely architectural though choice language obviously impact terms involving libraries language remains nutshell application contains objects representing memberships belonging people members fitness facilities memberships contain series recurring payments memberships automatically renew term others example membership initial period year renews application creating membership kind causes recurring payments created when last month expires does membership daily cron task responsible causing memberships expire based completed payments membership automatically renew cron task will renew memberships initial term simply user similar manner minus initial payment good makes things complicated additional freeze memberships hold specific durations automatically reactivate reappear people away vacation period time choose freeze membership right reactivate later choose schedule freeze setting freeze date point future reactivation date note always reactivation date makes things cancel memberships either right setting cancellation occur future future cancellations refund memberships cancellation except past payments makes difficult deal effect recurring payments when freeze membership recurring payments stretch around freezes period period time represents freeze paid conceptually programmatically difficult handle payments example extend different periods payment someone pays every week pays weeks membership date cancellation anywhere within period payment freezes taken approach membership object contains dates namely handles freeze period however client wants future cancellations noticed bugs freezing functionality leads believe need reconsider considering changing things future events scheduled effect recurring payments portion application idea queue particular events example freeze future accomplished queuing freeze event particular date that event subsequent date events connected single scheduled freeze users perspective future cancellation handled approach benefits example wanted cancel future cancellation kind annoying tricky stuff talking simply remove scheduled cancellation events nagging feeling simply jumping trying fire wondering anyone provide guidance issue design patterns existing architectural principles sort problem additional thing note recurring payments memberships scheduled terms automatically renewing exist database records deleted moved time price adjusted using temporal expressions martin fowler suggests appropriate problem know realize proposed solution events queue display user changes happen existing recurring payments think live code give creative respond great suggestions below comment boxes allow nearly level freeze period arbitrary length although practice imagine rare freeze weeks solution work regardless length renewal date changes pushed forward length freeze freeze weeks long pushes payment forward weeks things especially tricky businesses payments withdrawn specific dates example clubs process payments month when dates pushed around clubs snap particular explain detail rules affect event queuing management subscription interested amortization table concept basically exactly built already yearlong membership monthly payments creates weekly created amount associated along state machine governs pending paid failed refunded part struggling table responds events right freeze affects table immediately changing dates payments freeze middle table pushes payments forward sounds effective actually quite complex hard manage amortization table idea improve sounds event queue proposed originally seems obvious worked stuff impressed error check processing date great idea intend implement asap hoping explain suggestion little wondering concept deal recurring payment situation described original question and freeze scheduled longer accurate schedule will change once freeze takes place scheduled right middle payments schedule payments remain unchanged until date freeze became current date time freeze instituted payments move strikes perhaps great simplify application wondering users perceive indicate	ruby-on-rails;ruby; design-patterns; datatime; architecture		block, side, leave, turn, screen, piece, composite, reuse, android, callback, fragment, communicate, trigger, push, app, stop, notification, decouple, begin, expression	https://stackoverflow.com/questions/42987416/design-pattern-to-handle-the-interaction-between-different-types-of-objects	Architecture / Reusability Design	3	yes	theoretical	long term	yes	
61	schedule payments looking when freeze scheduled longer accurate schedule will change once freeze takes place design pattern handle different types objects program class called ball enum fields called ball colors blue yellow ball types tennis soccer bowling ball object possible color program ball objects different color type combinations balls interact collisions when balls collide triggers interaction handled function called ball interaction balls depends type color ball example bowling ball hits ping pong ball ping pong destroyed bowling ball continues move original velocity however soccer ball collides bowling ball soccer ball bounces bowling ball bowling balls velocity reduced example tennis ball ball function handling interactions using nested switch statements however write code handle scenarios nested switch statements seem right anyone think better handle interaction balls	java;oop; design-patterns; software-design		block, side, leave, turn, screen, piece, composite, reuse, android, callback, fragment, communicate, trigger, push, app, stop, notification, decouple, begin, expression	https://stackoverflow.com/questions/4195037/difficult-architectural-problem-involving-recurring-payments-and-future-events	Architecture / Reusability Design	2	yes	theoretical	long term	yes	
62	change peripheral timer embedded easy working embedded project microchip three timers driver lets call driver uses timer timer want driver program timer timer already driver rewrite driver makes timer timer driver maybe upcoming project want driver timer timer used driver having rewrite driver three timers possible combinations even basic different versions working driver making impractical manage being bugs code imagine driver timers even possible ways write driver solve problem write driver hardcoded timer assign suitable timer initialisation think every timer object inherits timer superclass pass object constructor working therefore cant objects	cdesign-patterns; embeds		block, side, leave, turn, screen, piece, composite, reuse, android, callback, fragment, communicate, trigger, push, app, stop, notification, decouple, begin, expression	https://stackoverflow.com/questions/51466832/change-peripheral-timer-in-embedded-software-easy	Architecture / Reusability Design	1	no	theoretical	long term	yes	
63	design break dependency classes using interface apologize advance will stuck trying learn unit testing design patterns once maybe problem reading unit testing clean code martin head first design patterns just beginning understand delegates events troll questions recent quite lambdas contextualize given myself learning project calling alarm class members expect name event trigger wanted timer alarm extensible created interface follows interface define component will raise alarm trigger will bubble alarm class raise trigger event alarm itself essentially timer component found component perfectly suited created class implements fine problem alarm class want create different kinds alarm example already heartbeat alarm triggered every interval minutes triggered date time daily alarm triggered every perhaps others perfectly suited handled problem design problem want able instantiate alarm kind alarm class derived classes knowing anything quartz problem quartz awesome factories return just right setup triggers will needed alarm classes example quartz trigger using create trigger heartbeat alarm described above daily alarm guess indirectly directly want alarm classes able consume classes knowing anything know contradiction asking question thought putting delegate field alarm assigned method doing creating hard dependency alarm quartz want avoid unit testing purposes design purposes thought using switch type know design trying learn good guys awesome thanks advance answers	cdesign-patterns; interface		block, side, leave, turn, screen, piece, composite, reuse, android, callback, fragment, communicate, trigger, push, app, stop, notification, decouple, begin, expression	https://stackoverflow.com/questions/29544477/design-question-how-to-break-the-dependency-between-classes-using-an-inter	Architecture / Reusability Design	4	no	program related	long term	yes	
64	custom reusable code overview currently custom subclass implements custom logic sure view class best place code making swift uses subclass call represents sliding door response swipe gesture sliding animation open complete animation trying keep view controller light actual core animation code handles animations class view controller handles gesture checking matches gesture required open given door calls open method open method just sliding animation check will used future differentiate types doors okay animation logic view class first instinct animations specific class methods precedent animation handles classes themselves continue develop will adding door types animations fear will grow animation code point simply start making subclasses view animations handled view controller problem besides bloated want view controllers need duplicate code packaged animations need call open	ios;animation; design-patterns; core-animation; cocoa-design-patterns		block, side, leave, turn, screen, piece, composite, reuse, android, callback, fragment, communicate, trigger, push, app, stop, notification, decouple, begin, expression	https://stackoverflow.com/questions/36166602/where-should-custom-reusable-animation-code-go	Architecture / Reusability Design	1	yes	program related	long term	yes	
65	sending data multiple screens particular screen source recycler view butts screen takes next screen next screen single plain activity activity view pager fragment buttons takes next screen screen edit song class field problem confused whether send list next screen further next fragment next screens intent static access anywhere again parcel wrap unwrap send fragment wrap fragment send attached fragment long process afford access anywhere change list 66 screen secondly whole process cumbersome every time sending intent receiving intent	android;android-fragments;design-patterns		block, side, leave, turn, screen, piece, composite, reuse, android, callback, fragment, communicate, trigger, push, app, stop, notification, decouple, begin, expression	https://stackoverflow.com/questions/47753299/send-ng-data-in-through-multiple-screens	Architecture / Reusability Design	3	no	theoretical	long term	yes	

NO	Text	Tags	Dominant Topic	Topic Terms	URL	Label	No. of Answers (Number)	Is Best Answer Selected? (Yes/No)	Is the question Theoretical or Program related? (Theoretical/Programming)	Is the solution temporary or long term? (Temporary/Long Term)	Is the discussion related to design? (Yes/No)	Is label representative? (Yes/No)
64	data mapper implementation strategies almost finished data mapper point comes will illustrate ideas able find good articles informations topic maybe reinventing wheel sure just framework want learn doing lets look relationships general when domain class called company called address company class will something lets cases just display list companies address needed when someone looks details case data mapper simply loads lazily meaning will just fetch database will join address data general getter method every relationship case method takes company object looks address property null fetches corresponding address object database using mapper class address object assigns address object address property specified company data mapper allowed data cases need company object address object always display list together case fetches company objects does query join fields address object finally iterates record feeds objects corresponding values assigning address object company simple difference company multiple address objects lets look when time interested company data mapper just addresses property company object null addresses property array reference none multiple addresses know load lazily just null need addresses cases display list together addresses case things start really ugly first cant join address table fifty times every address object strongly believe impossible performance below zero when think further down road impossible load lazily case important true send queries address objects companies address object contains country state city road house number house business tower lots companies modern buildings anyone rent small show tower website companies share plans deal kind problem important probably bigger problem anyone knows good goes details solving implementing	design-patterns;oop;datamapper	12	aggregate	https://stackoverflow.com/questions/1971798/data-mapper-and-relationships-implementation-strategies	Design-Principles-Validation-+Violation Module Access and Violation	3	yes	theoretical	long term	yes	
67	happy link square rectangle does break solid several threads pertaining classic example design breaks principle square rectangle classes problem starts demonstrating square extends rectangle violates people solve heard ideas including having polygon base class extended rectangle square thought wrong direction wrong discussion square does extend rectangle square does capability properties rectangle square rectangle type rectangle given rectangle class exists tasked building square class reusing rectangle build square class protected property instance rectangle expose rectangle apply square expose properties methods square apply rectangle appear square class solution argument goes reason breaking case square extends rectangle problematic someone else adds method rectangle alters area rectangle factor lets call method signature implemented multiplying width factor argument states squares will grow square factor width height width grows square grows exactly true system actually work actually happens width grows factor area rectangle grows factor instances square longer actually squares case square extends rectangle width height longer version someone else method rectangle does break square nobody method rectangle instance square exposed square class someone dictates method added programmer adds square class simply exposing rectangle class assuming person does runs unit test class will discover just broke whenever running method test added stack tests reveal problem development general necessarily design flow system correct answer implement method square increasing length square root factor argument inanimate class does guarantee rectangle inside remains square width height disagree square does expose methods change either rectangle methods break none exposed personally fits real world square rectangle squares rectangles rectangles squares does solution break rules solid	php;oop;design-patterns;solid-principles	12	aggregate	https://stackoverflow.com/questions/4848896/square-rectangle-implementation-that-does-not-break-solid-lsp	Design-Principles-Validation-+Violation Module Access and Violation	1	no	theoretical	long term	yes	
68	class violating single principle even thought think understand single principle cohesion principle following questions causing confusion assume planet bird properties arbitrarily random class code within needs operates objects returned properties words planet bird properties belong states object reason class violating break changes planet bird instances propagate cohesion refers closely related methods class level variables class highly cohesive class methods class level variables used together accomplish specific task class cohesion functions randomly inserted class used accomplish variety different even though class contains random properties accomplishes just single specific task several closely related task cohesion even though specific task several dosely related tasks assume planet bird properties used methods instance accomplish specific task high cohesion even though conceptually properties belong thus better instead planet bird instances passed arguments methods operate trunk encapsulated bird planet inside worse private class three reasons fall three reasons question cars methods even operate properties thus changes planets birds public wont affect confusing chances least first question having modified modification planet bird instances exactly regardless whether planet bird instances contained aggregated second question methods operate properties order perform single specific task conceptually least somewhat related even second question class cohesion even though per single task using properties accomplish task	oop;design-patterns;single-responsibility-principle	12	aggregate	https://stackoverflow.com/questions/14324917/is-car-class-violating-single-responsibility-principle	Design-Principles-Validation-+Violation Module Access and Violation	3	yes	theoretical	long term	yes	
69	making static reference method class learning programming java course doing project program creates three types objects address date employee program stores data several employees displays data array type using four different classes address class date class employee class class creates address date employee classes package using eclipse point employee class create object type employee able address name using address name date place array comes play program prompts user number employees stored array creates employee size idea code employee array variables classes obtained employee number employee name first last address street address city state code code hire date year obtained second loop iterates employee displays problem having eclipse gives error methods when example return eclipse says cannot static reference method solution static tried problem making methods variables address employee date classes values holding meaning data will displayed mean below sample output made methods variables static text asterisks user employees will data stored employee employee first name employee last name employee enter street address enter name city enter character code state enter code finally month hire number employee name bobby address avenue employee employee number employee name bobby address avenue employee making variables methods static values locked shown makes program useless does anyone solution problem need display information employee referencing methods classes normally just create variables methods class called employee assignment instructions specify need individual classes	java;arrays;class;oop;class-design	12	aggregate	https://stackoverflow.com/questions/36346942/making-a-static-reference-to-a-non-static-method-in-another-class	Design-Principles-Validation-+Violation Module Access and Violation	1	yes	program related	long term	yes	
70	product relationships creating check process step involves product cases product configuration configurable option option group consist selected option none group consists multiple user remove options product example option group called option specific option option example case options belonging database option group specific options group groups dependency option group specific items filtered requirement target option group target dependency need worry options product option group pointing target product option example order allow option selected database product group windows option selected operating system option order allow option selected database product group either windows options selected operating system option above diagram product option dependency windows product options product option group either operating system options selected product option dependency windows product option product option group when windows operating system selected will store dependency mapping question code evolves store relationship dependency mapping product options groups main issues questioning aggregate manage store mapping aggregate delete stop deletions products example dependency operating system windows product allow removal dependencies done application service build transaction aggregate easier transaction management higher potential concurrency store mapping within aggregate however someone updated name description whilst user editing mapping data concurrency exception really sense mapping data fail someone updates name unrelated others situation best structure code above scenarios missing deeper insight stating aggregates redesigned will things think accessing inside outside forbidden design cant think model proposed proposed answer taking time help really neat concise coding style answer does raise further questions below barking wrong tree appreciate clarification descriptions dictionary used contain descriptions option description property part option mentioned option value case member called type value objects allowed identifying code option takes constructor list understand option exist part type requires member group type option allowed hold reference option inside different aggregate instance does violates rule holding references aggregate roots reference things persisted aggregate roots child entities whole object separately having member within aggregate root option code takes dependencies type options rehydrated repository entity contained within entity part aggregate root passed constructor	design-patterns;domain-driven-design	12	aggregate	https://stackoverflow.com/questions/19932771/product-configuration-relationships	Design-Principles-Validation-+Violation Module Access and Violation	1	yes	program related	long term	yes	
71	dependencies custom validator components using validator service container project full custom validator requires dependencies supplied constructor documentation describes method configuring validator service done successfully makes point using service specs guice service seems specific when using full framework validator magically loading custom validator when link validator service container components strategy having validator load customer validator	php;validation;dependency-injection;symfony	12	aggregate	https://stackoverflow.com/questions/4061866/how-to-configure-dependencies-on-custom-validator-with-symfony-components	Design-Principles-Validation-+Violation Module Access and Violation	1	yes	theoretical	long term	yes	
72	dependencies duck example strategy pattern duplicate reading strategy design pattern java problem duck class several types ducks code flying quacking behaviors ducks obviously cannot code flying behaviors duck class methods especially different kind ducks create behavior interface called fields different classes duck associate behaviors ducks understand done composition inheritance cant create subclasses implement methods quack assign appropriate behavior ducks multiple inheritance supported nonflying squeaking duck cannot extend something else doing fed confused motto fads	java;design-patterns;inheritance	13	virtual, style, normal	https://stackoverflow.com/questions/22816843/duck-example-for-strategy-pattern	Strategy Design Pattern	3	no	theoretical	long term	yes	
73	composition inheritance someone explain depth design pattern game trying develop social game werewolf game characters usable passive abilities during night player pick usable ability choose targets night abilities happen time next results abilities usable abilities include someone preventing someone using ability night preventing abilities used passive abilities immunity types usable looking design pattern architecture prevents interdependence different abilities sorry maybe generic question trying head around quite	design-patterns;architecture;software-design	13	virtual, style, normal	https://stackoverflow.com/questions/59145615/design-pattern-for-avoiding-social-deduction-game	Strategy Design Pattern	1	no	theoretical	long term	yes	

NO	Text	Tags	Dominant Topic	Topic Terms	URL	Label	No. of Answers (Number)	Is Best Answer Selected? (Yes/No)	Is the question Theoretical or Program related? (Theoretical/Programming)	Is the solution temporary or long term? (Temporary/Long Term)	Is the discussion related to design? (Yes/No)	Is label representative? (Yes/No)
75	design pattern when modelling something different properties design perform crud entity lets call report specific type represents kind information going inform type reports will contain certain data property others will contain data example report type will contain things reports type kind information report will button exported excel file words report need provide different implementation excel generator depending reports data report contains needs added report class property problem know model will need retrieve reports database need details appropriate report generic represents type object property type object report somehow external just know design pattern help getting head around kind think explained problem properly second restriction know type report dealing storing reports database schema storing objects report property anything object want retrieve report certain want able cannot really simply database record appropriate report subtype contain excel generator assumes know already external report service depending type uses generator means time modify support wanted something respectful main problem something generic store data report depends know model	cd/design-patterns		behavior, inheritance, modify, compile, replace, override, delegate, advantage, prefer, setter, creation, compiler, extra, extension, enum, prevent, abstract, virtual, style, normal	https://stackoverflow.com/questions/4874947/which-design-pattern-to-use-when-modelling-something-that-can-contain-different	Strategy Design Pattern	1	no	program related	long term	yes	
76	file participate open closed principle know according open closed principle something unchanged when file designing game game monster information game client side download replace file monsters worrying format monster information file monster information monster individual monster information single file time monster will replace file monster individual time monster will monster start think case convenient when update monsters need replace case need know need extra store update status server suddenly remember open closed principle adding something modify current content seems reasonable replace file when monster above just example think file includes open closed principle asking format better main point want file format better definition open closed principle cannot find definition mentions file once confirmed deduce format file	file,design-patterns,language-agnostic,open-closed-principle		behavior, inheritance, modify, compile, replace, override, delegate, advantage, prefer, setter, creation, compiler, extra, extension, enum, prevent, abstract, virtual, style, normal	https://stackoverflow.com/questions/32047377/is-file-also-particulate-in-open-closed-principle	Strategy Design Pattern	0	no	program related	long term	yes	
77	dart patterns replace static inheritance dart static methods inherited primarily background static method inheritance possible made extremely easy thanks self static keywords writing dart keep finding myself running situation situation first thought need static method class classes example classes want able call create object data stored database classes store themselves database feel functionality inherited copy copy paste implementation every class store database time method does depend specific instance class indeed stretch think situations want grab data actually instantiating class begging static method dart wall unable inherit static method sure design patterns allow result static inheritance namely able reuse code multiple classes having create instance class when strictly needed dart will support idiomatic dart used something intended construct instance example above situations encountered relate construction played around factory constructors able work situations encountered problem doing async work ultimately need return future object cant constructor beyond trouble writing factory constructor superclass need access static fields target class something static keyword cant inject constructors call left having define static fields class without able even define common interface inherit asked concrete examples trying accomplish suppose making game different type upgrades purchased upgrade purchasable multiple times upgrade type class common interface define methods apply whatever along properties cost suppose want upgrade cost function basic cost type number active upgrades type already owned expensive purchase want adding static method superclass define standard function determines cost upgrade static method overridden specific upgrade type upgrade follow default price progression makes sense static context ten need able price type upgrade without actually needing instance upgrade type without static inheritance however either method mean need invoked need define static method every upgrade class former creates instances needed just throw away latter does nothing enforce existence method upgrade meaning able count kinda ruins polymorphism require copy paste logic multiple places solve java problem polluting namespace nothing enforce interface surely better example creating basic crud functionality objects define superclass work creating table display existing objects type form creating object static inheritance means cant next thought table building class gets objects passed want create object type reference class solution need pass table builder does will table building class need array items create function delete function update function passed constructor	design-patterns, dart		behavior, inheritance, modify, compile, replace, override, delegate, advantage, prefer, setter, creation, compiler, extra, extension, enum, prevent, abstract, virtual, style, normal	https://stackoverflow.com/questions/20726598/dart-patterns-to-replace-static-inheritance	Strategy Design Pattern	1	yes	theoretical	long term	yes	
78	java member method initializer pattern extend design correct safe improve avoid overridable method call dilemma delegate variables initialization calls overridable methods member method return reference instance self exception thrown trying instance invoking initialization methods creation objects will look super class implement methods interface methods initialize method write code following order call setter methods initialization call overridable method putting method overloaded provide versions necessary arguments illustrated complex example method check initialization done when call method class reference variable initialized initialize method initialization done throw subclass override initialize methods super class versions initialize method following order initialize call super method throw setter methods note super call setters avoid setter useless call super initialize method call super method throw overloaded versions initialize done calling parameters less parameter following code using complicated example using initializer using static method fact pattern expected static pattern ends final adapt pattern static failed know subclasses invoke superclass create method affect instance hope good idea wailing comments suggestions	java,design-patterns, constructor, initializer, overriding		behavior, inheritance, modify, compile, replace, override, delegate, advantage, prefer, setter, creation, compiler, extra, extension, enum, prevent, abstract, virtual, style, normal	https://stackoverflow.com/questions/20726598/java-member-method-initializer-pattern-to-what-extend-this-design-is-correct-bug	Strategy Design Pattern	1	no	program related	long term	yes	
79	visitor pattern java made research trying develop type framework provides ability convert instances source class instances result classes framework provide ability different conversion logic different converters pair source result extendable allow adding converters existing pairs source result requirement attempt convert instance source class instance result class converter implementing appropriate conversion logic lead compile time decided visitor pattern converters visitors classes elements provide extensibility decided generics first implementation conversion framework made influenced article internet unfortunately lost link framework core interfaces framework converter implementation accept implementations converter convert implementation converter visit implementations made convert example classes converted result classes code tests output built look result field main drawback implementation makes converters always handy trying avoid drawback framework double main point implementation parametrize converters result classes return results convert method converter method looks removed declaration having result class converter declaration fact parametrize implementations result classes bound implementation result class converted refers received converters interface problem implementations invoking received converter will invoke convert defined converter interface convert methods override converter implementations words will called basically kills main notion visitor pattern double dispatching test output demonstrates overriding methods called implementation tried stateless converters parametrized main notion parametrize methods want return conversion result without parametrizing declarations first glance looks great fact solution example following cause compile time error lead general question stateless visitor added links sources three implementations	java,generics, design-patterns, visitor-pattern		pointer, memory, accept, node, parse, free, session, hierarchy, alternative, visitor, hold, construct, usage, cast, graph, convert, calculate, extract, safe, stream	https://stackoverflow.com/questions/36383142/generics-implementation-of-visitor-pattern-in-java	Design Strategy Recommendations	2	no	theoretical	long term	yes	
80	grammar transformed parsing table unambiguous know general undecidable whether grammar unambiguous however does imply cannot decided subset grammar use transform input text parse tree grammar ambiguous produce parse tree given parser algorithm first trans grammar parser table afterwards uses parser automation process given input stream parse tree using parser table first step usually done parser generator second step executed parsing parser table construction algorithm error algorithm receives grammar table construction successful parser table returned table construction failed error returned examples algorithm typical examples algorithm failing finite input stream parser table parser automation deterministically derive parse tree given input stream return error input stream does match grammar parsing step formalized error parsing table input stream tokens single parse tree error returned input stream does match statements above understanding grammar somehow transformed parser table unambiguous however algorithm returns error does imply statement whether grammar ambiguous thus table construction algorithm whether subset grammars unambiguous steps fall knowledge none above true common table construction able find clear statement greatly appreciate explanation reference question discussed clearly think question quite relevant when designing programming language correct parser makes sure every program language properly parsed methods	design-patterns, tree,parallel, processing, algorithm,tree-traversal		pointer, memory, accept, node, parse, free, session, hierarchy, alternative, visitor, hold, construct, usage, cast, graph, convert, calculate, extract, safe, stream	https://stackoverflow.com/questions/36383142/generics-implementation-of-visitor-pattern-in-java	Design Strategy Recommendations	1	yes	theoretical	long term	yes	
81	ensure grammar unambiguous strategy implement tree traversing algorithm parallel implemented iterative algorithm involves tree traversal sometimes called downwards followed tree traversal upwards accumulation visit node involves calculating storing information used next visit either subsequent traversal subsequent iteration during traversal node processed independently long nodes not already processed processing node needs pass tuple specifically floats children traversal node processed independently long subtrees already processed processing node needs pass single float structure trees static unchanged during algorithm however during course downward traversal floats passed become zero entire subtree node does need processed upwards traversal node begin subtree preserved passed floats subsequent iterations become nonzero node traversals intensely computation node across tree computation node trivial just sums list numbers length equal number children trees processed unbalanced typical node leaves plus additional child nodes simply partitioning tree relatively balanced subtrees further trees designed consume available bigger tree process attend order iterations second just little test trees real trees expect slow order magnitude given algorithm requires less million iterations possibly billion reach acceptable result parallelize algorithm advantage multiple cores zero experience parallel programming recommended pattern parallelization given nature algorithm	design-patterns, tree,parallel, processing, algorithm,tree-traversal		pointer, memory, accept, node, parse, free, session, hierarchy, alternative, visitor, hold, construct, usage, cast, graph, convert, calculate, extract, safe, stream	https://stackoverflow.com/questions/2228855/strategies-to-implement-tree-traversing-algorithm-in-parallel	Design Strategy Recommendations	3	yes	theoretical	long term	yes	

