

## 1. Redogöra för objektorienterade designprinciper:

Redogör för några av SOLID-principerna, samt för någon av de övriga principerna vi tagit upp under kursen:

1. Vad säger principen?
  2. Ge något konkret exempel på hur man bryter mot, eller följer, principen.
  3. Vad fyller principen för syfte? Varför är det bra att följa principen?
- SOLID-principerna är:

Single Responsibility Principle	Open-Closed Principle
Liskov Substitution Principle	Interface Segregation Principle
Dependency Inversion Principle	
  - De andra principerna är:

Separation of Concern	Law of Demeter
High Cohesion, Low Coupling	Command-Query Separation Principle

## 2. Designmönster, samt deras syfte och effekt:

1. Vad är ett design pattern?
2. Varför använder man design patterns?
3. För några av följande design patterns:

Observer Pattern	Iterator Pattern	Factory Pattern
MVC	Template Method	Singleton
Chain of Responsibility		

- a. Vad är pattern P för något?
- b. När och hur kan man använda det?
- c. Vilka designproblem löser man genom att använda pattern P?

## 3. Grundläggande objektorienterade koncept

- Redogör för några av begreppen eller begreppsparen:

Klass och objekt	Statisk och dynamisk typ
Inkapsling	Arv, Komposition
Subklass, Superklass	Exceptions
Statisk & Icke-statisk metod	Statisk & Icke-statisk variabel
Konstruktör	Variabel och attribut
Primitiv typ	Referenstyp, Alias
Abstrakt klass, Interface	Overloading, Overriding
Refactoring	Trådar, Trådsäkerhet
Generisk klass/metod	Polymorfism

## 4. Redogör för laborationerna

1. Utgå från UML-diagrammet för den kod ni har skrivit och diskutera följande:
  - a. Finns det några ställen där ni följer en designprincip? *Var/Varför?*
  - b. Finns det några ställen där ni bryter mot en designprincip? *Var/Varför?*
  - c. Finns det några ställen där ni använder ett designmönster? *Var/Hur?*
  - d. Finns det några ställen där ni skulle kunna tillämpa ett designmönster?  
*Var och till vilket syfte?*
2. Redogör följande för er kodbas:
  - a. Kunna förklara hur ni har implementerat MVC i ert projekt och hur ni har tänkt kring er implementation.
  - b. Kunna förklara hur MVC kommunicerar mellan varandra i ert projekt och vad som händer var.
  - c. Kunna redogöra för händelseförloppet när man klickar på något i ert gränssnitt.
3. Redogöra för vad Unit Testing är och hur man kan åstadkomma det i Java.
4. Kunna redogöra för samt visa i koden hur ni har löst en specifik del av någon laboration.

OBS! Med reservation för ändringar