

CS314, Colorado State University
Software Engineering
Notes 4: Design & Architecture for OO
Software
Extra Stuff

James M. Bieman

CS 314 Colorado State Univ.

4x-1

Interview Questions on Design Patterns

1. Define creational design patterns?

Hint: Factory and Abstract Factory are creational design patterns.

Answer: Creational patterns are used to define and describe how objects are created at class instantiation time. Usually an abstract super class contains the details of the classes that are instantiated and the client class is unaware of such details. Singleton pattern, factory method pattern, abstract factory pattern are examples of creational design pattern.

CS 314 Colorado State Univ.

4x-2

Questions (continued)

2. A class instance can be created using the *new* operator. Why should we use creational design patterns to create objects?

Answer: Using *new* to create objects is a valid approach, but it's like hard coding the object type. If we're 100% sure that our object will be of the same class all the time, then we use *new* to create an object. When the nature of the object can change, we use creational patterns, which offer flexibility.

CS 314 Colorado State Univ.

4x-3

More Questions

3. Which object-oriented construct is used by the creational patterns to instantiate objects?

Answer: They use inheritance to determine the type of object that is instantiated.

CS 314 Colorado State Univ.

4x-4

Another One

4. When used by software developers, what is meant by the term *factory*?

Answer: A *factory* is a software mechanism for constructing objects. Factories are used to separate the creation of objects from their use. If you use factory methods for object creation, then new derived types (AKA subclasses) will not require changes in classes that use factories.

CS 314 Colorado State Univ.

4x-5

More

5. When should you use a Factory Pattern?

Answer: Use a Factory Pattern when a class does not know what class of objects it will need to create. A Factory Pattern can be used when we need to create an object of any one of a set of subclasses.

CS 314 Colorado State Univ.

4x-6

Factory Method versus Abstract Factory

- Factory Method uses an argument to determine the class of object to construct.
- Abstract Factory uses inheritance, usually with factory methods.

CS 314 Colorado State Univ.

4x-7

Example Abstract Factory (with Factory Methods)

```
1. public class MazeFactory {
2.     public Maze makeMaze() {return new Maze();}
3.     public Room makeRoom(int n) {return new Room(n);}
4.     public Wall makeWall() {return new Wall();}
5.     public Door makeDoor(Room r1, Room r2) {
6.         return new Door(r1, r2);}
7. }
8. public class EnchantedMazeFactory extends MazeFactory {
9.     public Room makeRoom(int n) {return new EnchantedRoom(n);}
10.    public Wall makeWall() {return new EnchantedWall();}
11.    public Door makeDoor(Room r1, Room r2)
12.    {return new EnchantedDoor(r1, r2);}
13. }
```

CS 314 Colorado State Univ.

4x-8

Singleton Pattern Allows only one instance of a class

```
1. public class SingletonObjectTest {
2.     private static SingletonObject singletonObject;
3.     private SingletonObjectTest() {
4.     }
5.     public static SingletonObjectTest getSingletonObject() {
6.         if (singletonObject == null) {
7.             singletonObject = new SingletonObjectTest();
8.         }
9.         return singletonObject;
10.    }
11. }
```

CS 314 Colorado State Univ.

4x-9

Source

<http://www.geekinterview.com/Interview-Questions/J2EE/Java-Patterns>

CS 314 Colorado State Univ.

4x-10