

Creational Design Patterns

- Direct constructor calls may suffice for simple cases:

```
Fred f=new Fred(); // Instance of class Fred
```

- However, you may use more elaborate creational patterns to:
 - Select between several sub-classes, each optimized for different conditions.
 - Configure objects as they are created.
 - Use pooling.

> Factory



- > A **central** point for instance creation.
May select which sub-class to create according to conditions.

> Abstract Factory

- > Select between several possible factories, where each factory generates its own **family** of objects.

> **Builder**

- > When constructing complex objects with multiple parts, separates the construction algorithm from the internal representation.

> **Prototype**

- > Clone an existing instance.

> **Singleton**

- > A class of which there can be **at most one** instance.

