Video Name: Abstract Class

Topics:

- purpose
- abstract class/method syntax
- @Override

Java Class(es): chess package. ChessPiece, Queen

Concept is same as C++. Use when have “is-a” relationship, where there is some common functionality and/or variables across all child classes, but also some behavior that must be defined for the child class. Example: DrawObject. All will have location (with getters/setters). There is also a draw method, but the behavior of the draw must be defined by the child (e.g., draw a Line, draw a Rectangle... can’t draw unless you know the type). So draw would be an abstract method of the parent class.

Abstract methods can only be contained in an abstract class:

```java
abstract public class ChessPiece {
    abstract public void move();
}
```

Every child class must define the abstract methods (unless child is also abstract). Eclipse will add the method skeletons.

@override. Message to compiler that this method is overriding a parent class function. Ensures that function signature (name, parameters) match. Very helpful, as it’s easy to think you’re overriding a method, but if there’s an error in the signature, you aren’t really (so child method will never be called).

Cannot instantiate an abstract class. But it’s common to declare a collection using the abstract class, then fill it with child objects.

```java
ChessPiece piece = new Queen(1,1);
//Can’t do this!
ChessPiece piece = new ChessPiece(1,1);
```