Abstract Classes, Abstract Methods

- to prevent us from making instances of a class we apply the abstract keyword to the class
- abstract classes are often ones that are purely conceptual without any instances (e.g. a generic Shape, an AbstractRobot)

```
abstract class AbstractRobot extends Robot {
  abstract void greet(AbstractRobot other);
  abstract void greet(TranslationRobot other);
  abstract void greet(CarrierRobot other);
}

abstract methods provide no implementation in the class, however, sub-classes may provide implementations
```

- usually an abstract class contains abstract methods, that is methods which are declared, but supply no implementation (any non-abstract sub-class is forced to implement all these methods)
- a class with one or more abstract methods must be declared abstract itself