1) **What are the two boolean literals? Which method of the Bug class uses boolean literals? Which method of the Location class uses string concatenation?**

- **Boolean literals:** true and false
- **The canMove() method of Bug uses Boolean literals.**
- **The getAdjacentLocation() of Location uses string concatenation.**

2) **The Location class declares a number of constants. Which reserved word identifies these variables as constants? What are their names and types? Why are constants used here instead of variables?**

- **Reserved word:** final
- **Constants in Location class:**
  - `public static final int LEFT = -90;`
  - `public static final int RIGHT = 90;`
  - `public static final int HALF_LEFT = -45;`
  - `public static final int HALF_RIGHT = 45;`
  - `public static final int FULL_CIRCLE = 360;`
  - `public static final int HALF_CIRCLE = 180;`
  - `public static final int AHEAD = 0;`
  - `public static final int NORTH = 0;`
  - `public static final int NORTHEAST = 45;`
  - `public static final int EAST = 90;`
  - `public static final int SOUTHEAST = 135;`
  - `public static final int SOUTH = 180;`
  - `public static final int SOUTHWEST = 225;`
  - `public static final int WEST = 270;`
  - `public static final int NORTHWEST = 315;`
- **Constants cannot be changed. These constants represent degree measures that should not change.**

3) **In the four lines of code from the BugRunner program, how many different objects are created? How would you change the code so that the bug and rock are assigned to variables before being added to the world? Why don’t they need to be assigned to variables?**

- **Three objects are created in the first four lines of code in BugRunner:**
  - ActorWorld world = new ActorWorld(); //new ActorWorld object
  - world.add(new Bug()); //new Bug object
  - world.add(new Rock()); //new Rock object
- **Code modification:**

```java
ActorWorld world = new ActorWorld();
Bug ant = new Bug();
Rock stone = new Rock();
world.add(ant);
world.add(stone);
world.show();
```

This is not necessary, because by using the statement `new Bug()` as a parameter to the `add()` method, for example, the object is created and sent as a parameter at the same time.
4) In the first line of `Location.getAdjacentLocation`, shown above, suppose direction has the value 90. What value will be assigned to `adjustedDirection`?

First line: `int adjustedDirection = (direction + HALF_RIGHT / 2) % FULL_CIRCLE;`

Value assigned:
- `direction = 90`, `HALF_RIGHT = 45`, `FULL_CIRCLE = 360`
- `(90 + 45 / 2) % 360 = 112`