ROBOT APPLICATION

OBJECTIVE

- Extensible – easily adaptable to future enhancements
- Highly cohesive and loosely coupled
- Good use of OOPS concepts
- Make use of design pattern wherever possible
- Code should be clean. Avoid code smells.
- Proper Exception handling
- Can easily be unit tested. If possible, write unit tests

PROBLEM STATEMENT

Time Duration: 2.5hrs

A company working in artificial intelligence domain is planning to create a Robot. But before they do heavy investment in Robot research and development, they requested their technical team to do a small prototype and create a Robot with some basic features.

Technical team decided to introduce following features in the Robot prototype:

Power Operated:

1. Robot works on battery and can walk for 5 km per charge.
2. If remaining battery is less than 15%, a red light on Robot head should lit up indicating low battery.

Handling physical objects:

3. Robot can carry any object not weighing more than 10 Kg.
4. For every Kilogram carried by Robot, 2% extra [in addition to walking discharge] battery will be consumed.
5. If the weight of the object is more than 10 Kg, Robot display [LED display on chest] will show message “Overweight”.

Technical team handed over these details to IT team to build a software for Robot. Please create a Robot application in Java for automating all the features listed above.

Create Junits to test these functionalities. Also, please provide out of how much battery is remaining in following scenarios:

Robot walks for 3.5 KM	
Robot walks for 2 Km carrying 3 Kg weight	
Robot carries 12 Kg weight.	

Following are acceptable assumptions:

1. Display Message can be programmed as System.out.println.
2. There is no need for creating bar code scanning functionality. We can assume that we already have an api which does that for us.