Problem Statement: Junit testing and Design Principals

Create below maven Java project, add Junit dependency to the pom file.

1. Create a Junit test to test the methods of the class Number
2. Methods are
   1. Int checkPrime(int x)
   2. Boolean checkAmstrong(int x)
   3. Boolean CheckPolindrom(int x)
3. Create a test case for a functional, and non-functional check.
4. Create a class Employee POJO which has id, name, and city
5. Create a class AddEmployee which has a method
   1. List<Employee> list = new ArrayList()// this is to store the empoyees
   2. Void AddEmployee(Employee emp); // this method will add a new employee
   3. Employee getEmployee(int id)// will return the employee base on the id if the employee is available else returns null.
6. Create the test case to check the above AddEmployee class.
7. Create a calculator demonstration which has
   1. **S**ingle Responsibility Principle
   2. [**L**iskov Substitution Principle](https://medium.com/@severinperez/making-the-most-of-polymorphism-with-the-liskov-substitution-principle-e22609866429)