TDD – DESENVOLVIMENTO ORIENTADO A TESTES.

# 1 - GERÊNCIA DE CONFIGURAÇÃO

Passo 1 – Instalar sdk Java 7.

O download da JDK pode ser feito neste link:

<http://www.oracle.com/technetwork/pt/java/javase/downloads/jdk7-downloads-1880260.html>

Passo 2 – Baixar o Eclipse para criarmos o projeto Java.

O download da IDE Eclipse pode ser feito no seguinte link:

<https://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/kepler/SR2/eclipse-standard-kepler-SR2-win32.zip>

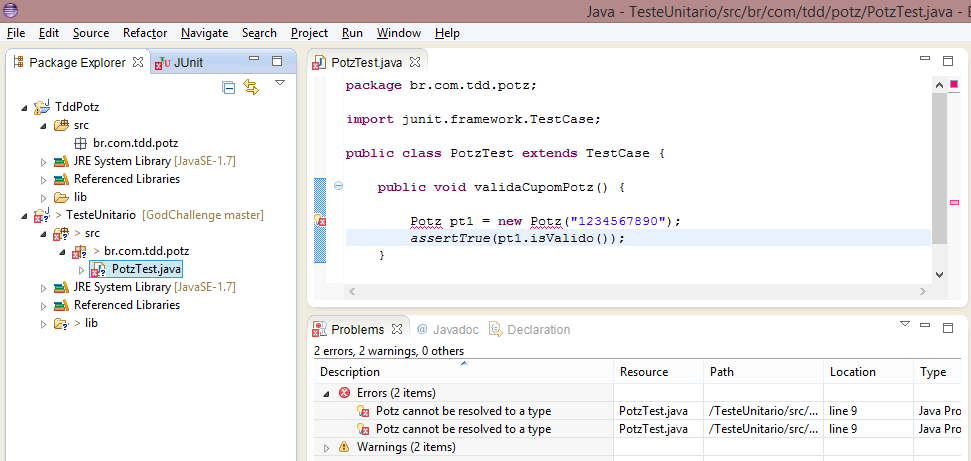
ou

<https://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/kepler/SR2/eclipse-standard-kepler-SR2-win32-x86_64.zip>

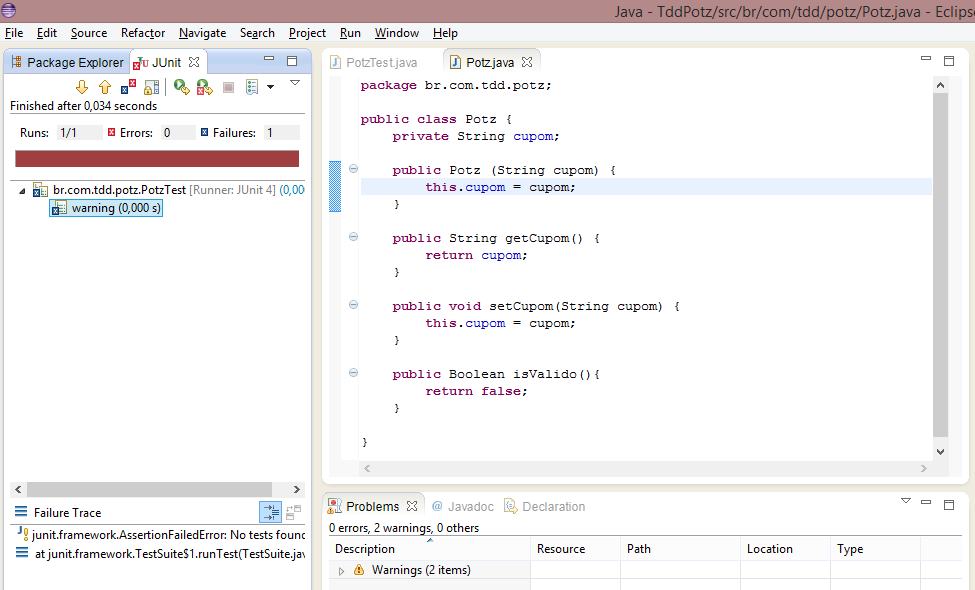
O eclipse não é necessário fazer a instalação, basta descompactar o arquivo zip em um diretório qualquer e executar o arquivo eclipse.exe.

# 2 – CRIAÇÃO DO PROJETO

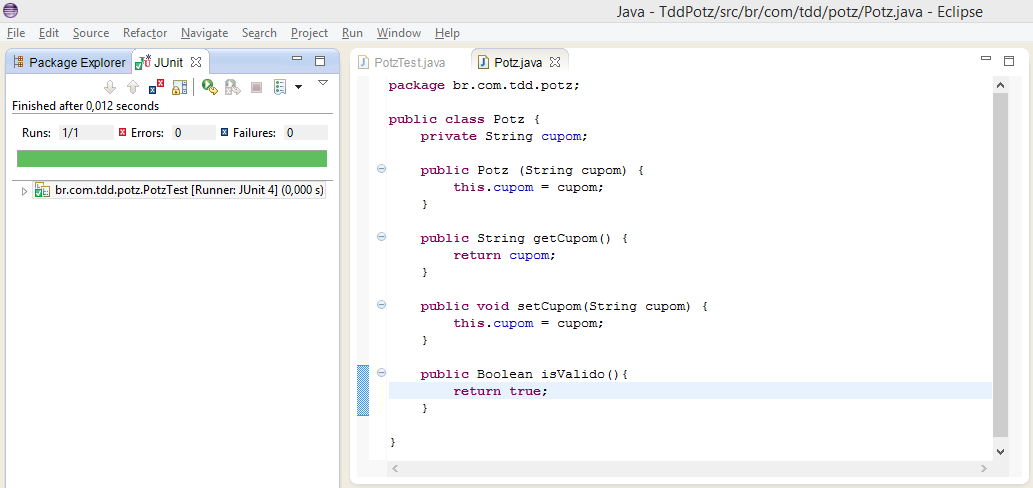
Passo 1 Tdd – Criação do primeiro teste – Erro Classe não existe



Passo 2 Tdd – implementada a Classe Potz e o projeto agora compila mas não passa no teste

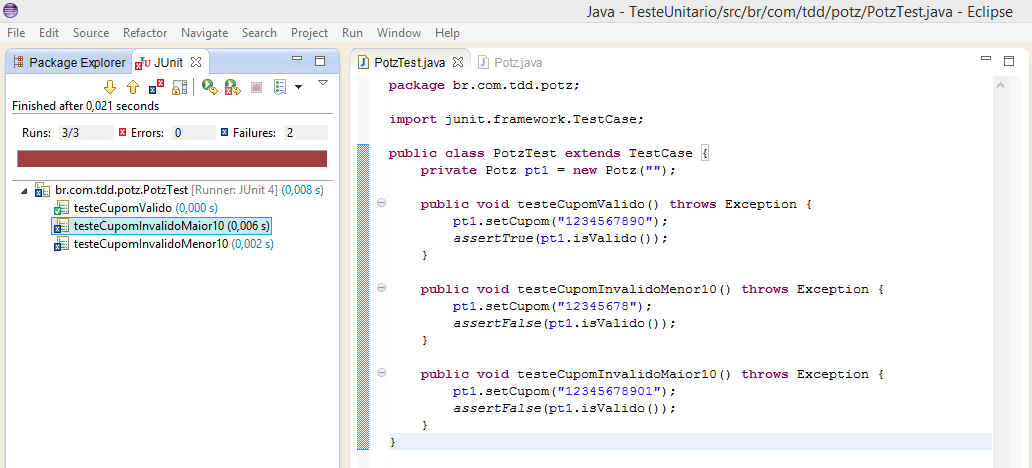


Passo 3 – Fazer o caso de teste passar no teste o mais rápido possível

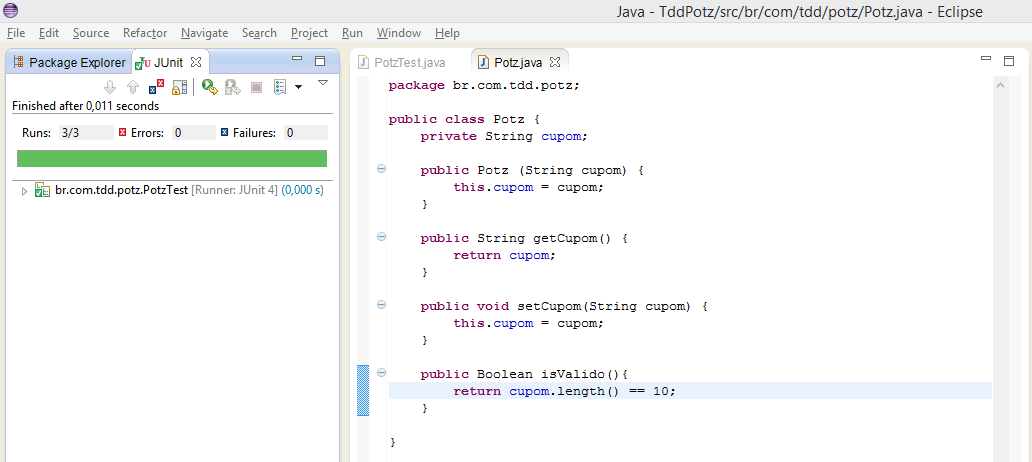


Criação de 2 novos casos de teste para verificar se o cupom é válido (tamnho = 10)

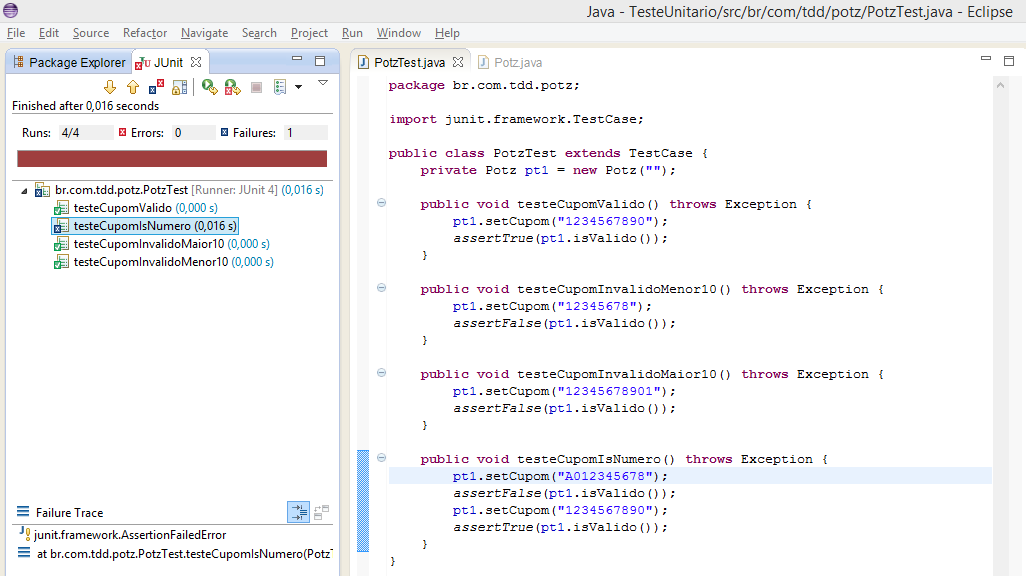
O caso de teste apresenta 2 erros.



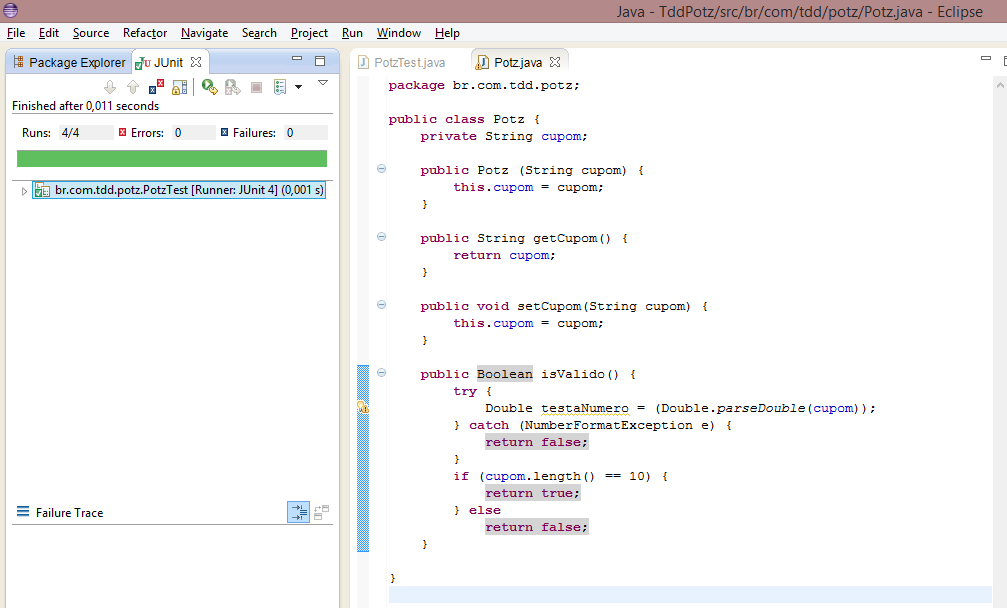
Correção da função is valido. Agora todos os teste passam



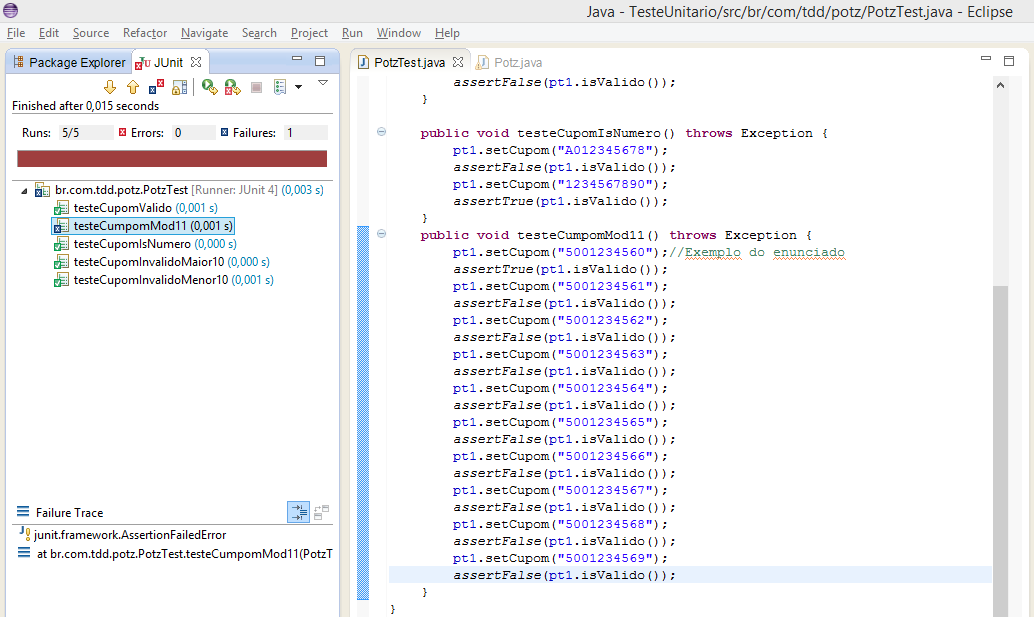
Criação de um novo caso de teste para verificar se o cupom é número: caso de teste não passa



Alterada a função isValido(), agora todos os teste passam



Criado um caso de teste para verificar se o cupom tem digito verificador mod11 válido. Teste não passa



Criado um caso de teste para uma função utilitária que irá verificar o dígito verificador.

Caso de teste não compila pois a classe ainda não foi implementada.

