

# Rapport TP6

## 1. Test de création d'un pokemon Aquali:

```
@Test
public void createAquali() throws PokedexException {
    double delta = 0.0001;
    // Test avec PokemonFactory
    Assert.assertEquals(aquali.getCp(),factory.createPokemon(0,2729, 202,5000, 4).getCp());
    Assert.assertEquals(aquali.getHp(),factory.createPokemon(0,2729, 202,5000, 4).getHp());
    Assert.assertEquals(aquali.getIndex(),factory.createPokemon(0,2729, 202,5000, 4).getIndex());
    Assert.assertEquals(aquali.getCandy(),factory.createPokemon(0,2729, 202,5000, 4).getCandy());
    Assert.assertEquals(aquali.getDust(),factory.createPokemon(0,2729, 202,5000, 4).getDust());
    Assert.assertEquals(aquali.getIv(),factory.createPokemon(0,2729, 202,5000, 4).getIv(),delta);

    // Test avec RocketFactory
    Assert.assertEquals(aquali.getCp(),rocketPokemonFactory.createPokemon(0,2729, 202,5000, 4).getCp());
    Assert.assertEquals(aquali.getHp(),rocketPokemonFactory.createPokemon(0,2729, 202,5000, 4).getHp());
    Assert.assertEquals(aquali.getIndex(),rocketPokemonFactory.createPokemon(0,2729, 202,5000, 4).getIndex());
    Assert.assertEquals(aquali.getCandy(),rocketPokemonFactory.createPokemon(0,2729, 202,5000, 4).getCandy());
    Assert.assertEquals(aquali.getDust(),rocketPokemonFactory.createPokemon(0,2729, 202,5000, 4).getDust());
}
```

- Échec du test de récupération d'IV. En effet, dans l'implémentation RocketPokemonFactory IV peut avoir deux valeurs soit 1 ou 0 .Ce qui explique pourquoi le test ne réussit pas tandis que les tests de mon implémentation réussissent.
- Essais de récupération HP, CP, Index, Dust, Candy pass . Parce que, dans l'implémentation de RocketPokemonFactory, le pokémon est créé à partir de ces valeurs . même mes tests passent.
- Les tests pour récupérer Attack, Defense, Stamina ne passent pas non plus car ces valeurs sont générées aléatoirement. Par contre, les tests pour mon implémentation sont réussis.

## 2. Test de récupération un pokémon avec un index invalide:

```
@Test
public void Test_GetPokemon_with_invalidIndex() throws PokedexException {
    //assertThrows(PokedexException.class, () -> pokemonMetadataProvider.getPokemonMetadata(-12));
    //assertThrows(PokedexException.class, () -> pokemonMetadataProvider.getPokemonMetadata(189));

    // Test avec PokemonFactory
    assertNull(factory.createPokemon(151,2729, 202,5000, 4));

    //Test qui passe pas avec RocketFactory
    //assertNull(rocketPokemonFactory.createPokemon(151,2729, 202,5000, 4));
};
```

- **Mon implémentation retourne null et crée une exception. Cependant, l'implémentation de RocketFactory ne génère pas d'exceptions et accepte l'index non valide. D'où l'échec des tests.**