

Evaluation

A series of overlapping wavy lines in orange, yellow, blue, and black, spanning the width of the slide below the title.

LP IEM

□ Gestionnaire de score d'un match de Tennis (Tests unitaires)

- Développer une classe TennisMatch permettant de gérer l'évolution du score d'un match de tennis en fonction du joueur qui marque un point
- La classe TennisMatch devra posséder au minimum les méthodes suivantes

Constructeur de TennisMatch

- MatchType est un enum qui pourra prendre les valeurs BEST_OF_THREE ou BEST_OF_FIVE (match au meilleur des 3 sets ou des 5 sets)
- Le constructeur attend un booléen pour savoir s'il y a un tie break dans le dernier set en cas d'égalité

TennisMatch
<ul style="list-style-type: none">• TennisMatch(Player, Player, MatchType, boolean)• updateWithPointWonBy(Player): void• pointsForPlayer(Player): String• currentSetNumber(): int• gamesInCurrentSetForPlayer(Player): int• gamesInSetForPlayer(int, Player): int• isFinished(): boolean

Player
<ul style="list-style-type: none">• Player(String)• getName(): String

- pointsForPlayer(Player): String retourne le score du joueur passé en argument dans le jeu en cours ("40", "15", "A" pour avantage)
- gamesInCurrentSetForPlayer(Player): int retourne le nombre de jeu gagné par le joueur passé en argument dans le set en cours
- gamesInSetForPlayer(int, Player): idem mais dans le set dont le numéro est passé en argument (le premier set est numéroté 1)

☐ Gestionnaire de score d'un match de Tennis (Tests unitaires)

- ☐ Les classes devront être accompagnées de test unitaires
 - ☐ Les tests devront être propres et lisibles
 - ☐ Introduire toute méthode permettant d'améliorer la lisibilité des tests et d'alléger leur écriture
 - ☐ Le respect des prototypes donnés est essentiel pour que les classes puissent être validées avec des tests unitaires écrits par d'autres
- ☐ La qualité du code est TRES IMPORTANTE
 - ☐ les variables et méthodes doivent être judicieusement nommées
 - ☐ méthodes et classes cohérentes
 - ☐ Eviter le code dupliqué

□ Gestionnaire de score d'un match de Tennis (Prototypes)

```
public TennisMatch(Player player1, Player player2, MatchType matchType, boolean tieBreakInLastSet)
public void updateWithPointWonBy(Player player)
public String pointsForPlayer(Player player)
public int currentSetNumber()
public int gamesInCurrentSetForPlayer(Player player)
public int gamesInSetForPlayer(int setNumber, Player player)
public boolean isFinished()
```

□ Exemple d'enum pour le type de match

```
public enum MatchType {
    BEST_OF_THREE(2), BEST_OF_FIVE(3);

    private final int numberOfSetsToWin;

    private MatchType(int numberOfSetsToWin) {
        this.numberOfSetsToWin = numberOfSetsToWin;
    }

    public int numberOfSetsToWin() {
        return numberOfSetsToWin;
    }

    public int maxNumberOfSets(){
        return numberOfSetsToWin*2 - 1;
    }
}
```