|  |  |  |
| --- | --- | --- |
| EXAMEN  Semestre : 1 2  Session : Principale Rattrapage | | logo-esprit |
| Code : | ETUDIANT(e)  Nom et Prénom : …………………………………………………...  Classe: ………………………………………………….................... | |
| Module : Conception orientée objet et programmation Java …………………………  Enseignant(s) : Equipe Java …………………………………  Classe(s) : 3INFO A, 3 SIGMA, 4INFINI ……………………………………… | | |
| Documents autorisés  : NON Nombre de pages : 8  Calculatrice autorisée  : NON Internet autorisée : NON | | |
| Date : 07/01/2017 ………………… Heure . 9:00................ Durée : 1:30h.................. | | |

**\*\*✂**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Observations** | **Nom et Signature du Correcteur** | **Nom et Signature du Surveillant** | **Note**  **/20** | **Code** |

Module :………………………………………………………………………………

**NB :** Les parties réservées aux informations relatives à l’étudiant et à l’administration seront ajoutées au cas où les réponses seront écrites sur les feuilles d’examen.

public class Joueur {

private int id;

private String nom;

private String nationalite;

public Joueur() { }

public int getId() { return id; }

public void setId(int id) { this.id = id; }

public String getNom() { return nom; }

public void setNom(String nom) { this.nom = nom; }

public String getNationalite() { return nationalite; }

public void setNationalite(String nationalite) { this.nationalite = nationalite; }

public String toString() {

return "Joueur{" + "id=" + id + ", nom=" + nom + ", nationalite=" + nationalite '}'; }

public int hashCode() { return id; }

public boolean equals(Object obj) {

if (obj instanceof Joueur) {

final Joueur other = (Joueur) obj;

return this.id != other.id;

}

return false;

} }

**NE RIEN ECRIRE**

**✂**

public class Competition {

private int id;

private String nom; }

public Competition() { }

public int getId() { return id; }

public void setId(int id) { this.id = id }

public String getNom() { return nom; }

public void setNom(String nom) {

this.nom = nom;

}

public String toString() {

return "Competition{" + "id=" + id + ", nom=" + nom + '}';

public int hashCode() { return id; }

public boolean equals(Object obj) {

if (obj instanceof Competition) {

final Competition other = (Competition) obj;

return this.id != other.id;

}

return false;

}

}

public class Equipe {

private int id;

private String nom;

private String pays;

private List<Joueur> joueurs;

public Equipe() { joueurs = new ArrayList<>() ; }

**NE RIEN ECRIRE**

**✂**

public int getId() { return id; }

public void setId(int id) { this.id = id; }

public String getNom() { return nom; }

public void setNom(String nom) { this.nom = nom; }

public String getPays() { return pays; }

public void setPays(String pays) { this.pays = pays; }

public List<Joueur> getJoueurs() { return joueurs; }

public void setJoueurs(List<Joueur> joueurs) { this.joueurs = joueurs; }

public String toString() {

return "Equipe{" + "id=" + id + ", nom=" + nom + ", pays=" + pays + ", joueurs=" + joueurs + '}'; }

public int hashCode() { return id; }

public boolean equals(Object obj) {

if (obj instanceof Equipe) {

final Equipe other = (Equipe) obj;

return this.id != other.id;

} return false; }

public void addJoueur(Joueur j) { joueurs.add(j); }

public void supprimerJoueur(Joueur j) { joueurs.remove(j); }

**NE RIEN ECRIRE**

**✂**

/\* TODO 1 : Avec l’api Stream \*/ (1 point)

public boolean rechercherJoueurParNationalite(String nationalite) {

return ……………………………………………………………………………………

………………………………………………………………………………………………… }

/\* TODO 2 : Avec l’api Stream \*/(1 point)

public Map<String, List<Joueur>> regrouperParNationalite() {

return ……………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

………………………………………………………………………………………………… }}

public class EnsembleEquipe {

public Set<Equipe> equipes;

public Set<Equipe> getEquipes() { return equipes; }

public void ajouterEquipe(Equipe e) { equipes.add(e); }

public void supprimerEquipe(Equipe e) { equipes.remove(e); }

/\* TODO 3 \*/ (0,5 point)

public boolean rechercherEquipe(Equipe e) {

return …………………………………………………………………………………………… }

|  |  |  |
| --- | --- | --- |
| EXAMEN  Semestre : 1 2  Session : Principale Rattrapage | | logo-esprit |
| Code : | ETUDIANT(e)  Nom et Prénom : …………………………………………………...  Classe: ………………………………………………….................... | |
| Module : Conception orientée objet et programmation Java …………………………  Enseignant(s) : Equipe Java …………………………………  Classe(s) : 3INFO A, 3 SIGMA, 4INFINI ……………………………………… | | |
| Documents autorisés  : NON Nombre de pages : 8  Calculatrice autorisée  : NON Internet autorisée : NON | | |
| Date : 07/01/2017 ………………… Heure . 9:00................ Durée : 1:30h.................. | | |

**\*\*✂**

/\* TODO 4 : Avec une méthode de référence ou une expression lambda \*/ (1 point)

public void afficherEquipes() {

………………………………………………………………………………………………………

……………………………………………………………………………………………………}

/\* TODO 5 : Avec lambda Expression ou l’api Stream \*/ (2 points)

public Set<Equipe> trierEquipesParNom() {

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

…………………………………………………………………………………………………………}

public class Fifa {

private Map<Competition, EnsembleEquipe> mapFifa;

public Fifa() { mapFifa = new HashMap<>(); }

/\* TODO 6 \*/ (0,5 point)

public void ajouterCompetition(Competition c) {

………………………………………………………………………………………………………

…………………………………………………………………………………………………… }

**NE RIEN ECRIRE**

**✂**

/\* TODO 7 \*/ public void ajouterEquipe(Competition c, Equipe e) { (2 points)

…………………………………………………………………………………………………………

…………………………………………………………………………………………… ……………

…………………………………………………………………………………………………………

………………………………………………………………………………………………………

…………………………………………………………………………………………… ……………

………………………………………………………………………………………………………… }

/\* TODO 8 Avec lambda expression\*/ (2 points)

public void afficherFifa() {

…………………………………………………………………………………………………………

………………………………………………………………………………………………………… }

/\* TODO 9 Avec l’api Stream\*/ (2 points)

public List<Equipe> retournerEquipesParPays(String nom) {

return …………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………

………………………………………………………………………………………………………}

**NE RIEN ECRIRE**

**✂**

/\* TODO 10 Avec l’api Stream\*/ (3 points)

public void afficherEquipesPersonnalisees(Competition c, String nationalite) {

……………………………………………………………………………………………………

…………………………………………………………………………………………………………

…………………………………………………………………………………………………………

…………………………………………………………………………………………………………

…………………………………………………………………………………………………………

…………………………………………………………………………………………………………

…………………………………………………………………………………………………………

…………………………………………………………………………………………………………

…………………………………………………………………………………………………………

……………………………………………………………………………………………………… …}

/\* TODO 11 \*/ public List<Competition> retournerCompetitionParJoueurTunisien() { (3 points)

………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

…………………………………………………………………………………………………………………………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

…………………………………………………………………………………………………………}

e

**NE RIEN ECRIRE**

**✂**

/\* TODO 12 \*/ (3 points)

public boolean returnSiUneEquipeParticpeEnPlusieursCompetitions() { ……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

……………………………………………………………………………………………………………

…………………………………………………………………………………………………………}}

Bon travail