

 <b>FACULTE DES SCIENCES BEN M'SICK</b> UNIVERSITÉ HASSAN II DE CASABLANCA	<b>Nom :</b> .....
	<b>Prénom :</b> .....
	<b>Salle :</b> .....
	<b>Code apogée :</b> .....

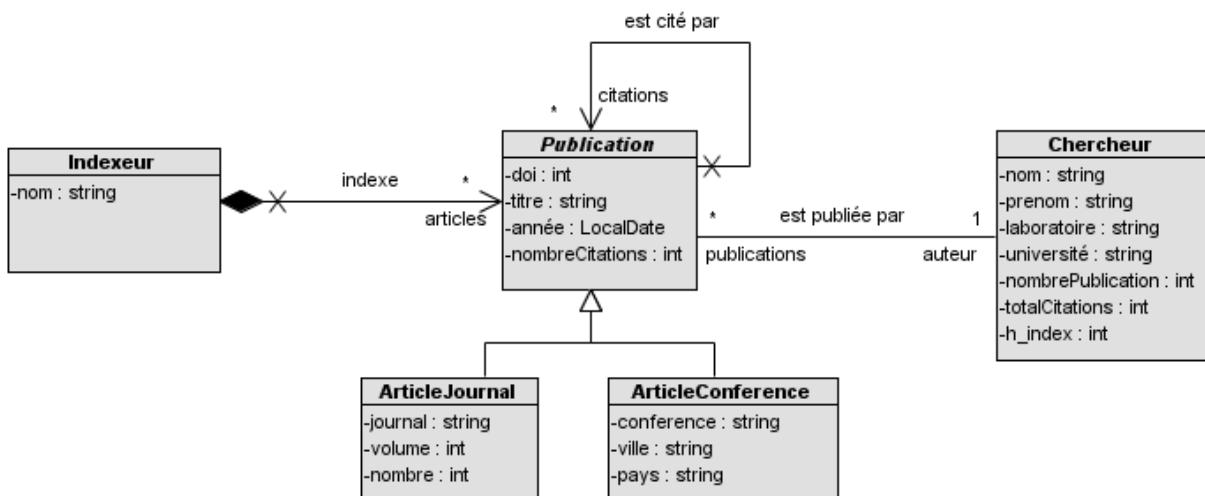
**Examen Java Session de Printemps 2024**  
**SMI S6**  
**Pr. Abdessamad Belangour**

---

Nous souhaitons réaliser une petite application très simplifiée de gestion des publications scientifiques d'un chercheur. Ainsi, lorsqu'un chercheur arrive à des résultats scientifiques importants, il rédige un article et le publie soit dans une conférence, soit dans un journal scientifique (revue scientifique). Pour simplifier nous allons considérer qu'une publication ne dispose que d'un seul auteur.

Pour qu'une publication soit le plus visible possible, des sites bibliographiques spéciaux (appelés Indexeurs) répertorient, organisent et indexent ces publications provenant de diverses conférences et journaux à travers le monde. Ainsi chaque chercheur est identifié et son travail est diffusé et évalué à travers un ensemble d'indicateurs (nombre de publications, nombre de citations, H-index et autres). Un chercheur à un h-index de 10, par exemple, si au moins 10 de ses publications ont été cités 10 fois.

Le diagramme de classe de cette application est comme suit :



Compléter le code qui manque :

```

public class Indexeur {
    private String nom;
    private Map<Integer, Publication> articles;
    public Indexeur(String nom) {
        this.nom = .....;
        this.articles = .....;
    }
    public Indexeur(String nom, Map<Integer, Publication> articles) { // fourni}
    public Publication recupererPublication(int doi) {
        .....;
    }
}
  
```

```

public boolean ajouterPublication(Publication p) {
    .....
    .....
    .....
    .....
}

@Override
public String toString() {
    return "Nom Indexeur=" + nom + ", Nombre articles=" + .....;
}
}

```

**// on suppose que les getters et setters sont fournis à part ceux qui sont demandés**

```

public abstract class Publication implements Comparable<Publication>{
    private int doi; // document object identifier
    private String titre;
    private LocalDate année;
    private Chercheur auteur;
    private int nombreCitations;
    private Map<Integer,Publication> citations;
    // nouvelle publication non encore citée
    public Publication(int doi, String titre, LocalDate année, Chercheur auteur) {
        this.doi = doi;
        this.titre = titre;
        this.année = année;
        this.auteur = auteur;
        this.nombreCitations = ..... ;
        this.citations=..... ;
    }
    // nouvelle publication citée
    public Publication(int doi, String titre, LocalDate année, Chercheur auteur, Map<Integer,
Publication> citations) {
        this.doi = doi;
        this.titre = titre;
        this.année = année;
        this.auteur = auteur;
        this.nombreCitations = ..... ;
        this.citations = ..... ;
    }
    public void setCitations(Map<Integer, Publication> citations) {
        ..... ;
        ..... ;
    }
    public boolean ajouterCitation(Publication pub){
        .....
        .....
        .....
        .....
    }
}

```

```

        .....
    }
    public boolean supprimerCitation(Publication pub){ //fournie en cas de besoin}
//retourne le titre l'année et le nombre de citations
@Override
public String toString() {
    .....
}

}
// comparer par nombre de citations, sinon par année sinon par titre
@Override
public int compareTo(Publication autrePublication) {
    .....
}
}
}

```

```

public class ArticleConference extends Publication {
    private String conference;
    private String ville;
    private String pays;
    public ArticleConference(String conference, String ville, String pays, int doi, String titre,
LocalDate année) { // fourni en cas de besoin}

    public ArticleConference(String conference, String ville, String pays, int doi, String titre,
LocalDate année, Chercheur auteur) {
        .....
    }

    public ArticleConference(String conference, String ville, String pays, int doi, String titre,
LocalDate année, Chercheur auteur, Map<Integer, Publication> citations) { // fourni }
//retourne le nom de la conférence, le titre l'année et le nombre de citations
@Override
public String toString() {
    .....
}
}

```

```
public class ArticleJournal extends Publication {  
    private String journal;  
    private int volume;  
    private int nombre;  
    // tout est fourni pour cette classe (Constructeurs, getters et setters, toString())  
}
```

