

# Session vs Cookie

- **SESSION** : les variables de session sont stockées sur **le serveur** **le temps** de la présence d'un visiteur sur le site
- **COOKIE** : les cookies sont enregistrés sur l'ordinateur du **visiteur** pendant **plusieurs mois**

# Session

- Une session est un mécanisme qui permet de conserver des données ou variables entre plusieurs requêtes d'un utilisateur sur un site web
- En PHP, une session peut être créée à l'aide de la fonction **session\_start()**
- Cette fonction doit être appelée au **début** de chaque script PHP qui utilise des variables de session
- Elle associe un **identifiant** de session unique à l'utilisateur
- Une fois que la session démarrée, on peut définir des variables de session à l'aide de la superglobale **\$\_SESSION**
- **Ex.** `$_SESSION['nom'] = 'Piere';`

# Fonctions de session

- **session\_start()**: démarre une session. Lorsqu'un visiteur arrive sur le site, un numéro ou ID de session est généré pour lui. C'est un nombre hexadécimal.
- Cette fonction est appelée au **début** des pages où on a besoin de conserver des variables de session
- **session\_destroy()**: ferme la session du visiteur
- Elle est automatiquement appelée lorsque un visiteur **se déconnecte** ou **ne charge plus** la page pendant plusieurs *minutes* (timeout)
- **session\_id()**: indique l'identifiant de la session
- **etc**

# Cookie

- Un cookie est un petit fichier que l'on enregistre sur l'ordinateur du **visiteur**
- Ce fichier contient du texte et permet de conserver des informations sur le visiteur (son pseudo, ses préférences, etc )
- Chaque cookie a une *date d'expiration*. Après cette date, il sera automatiquement supprimé par le navigateur
- Les cookies sont donc des informations **temporaires** stockées sur l'ordinateur des visiteurs

- Pour écrire un cookie, on utilise la fonction PHP **setcookie()** qui possède, en général, **trois** paramètres:
  1. le **nom**
  2. la **valeur**
  3. la **date d'expiration** (timestamp)
- Ces informations sont sauvegardées dans la superglobale **\$\_COOKIE**

# Modèle MVC

- Le modèle MVC (**Modèle-Vue-Contrôleur**) est un modèle de **conception** logicielle
- Il est souvent utilisé pour le développement d'applications **Web**
- Il permet la séparation de la **présentation**, la **logique** et les **données** de l'application
- Il permet de mieux **organiser**, **faciliter** la maintenance et **améliorer** la réutilisabilité du code

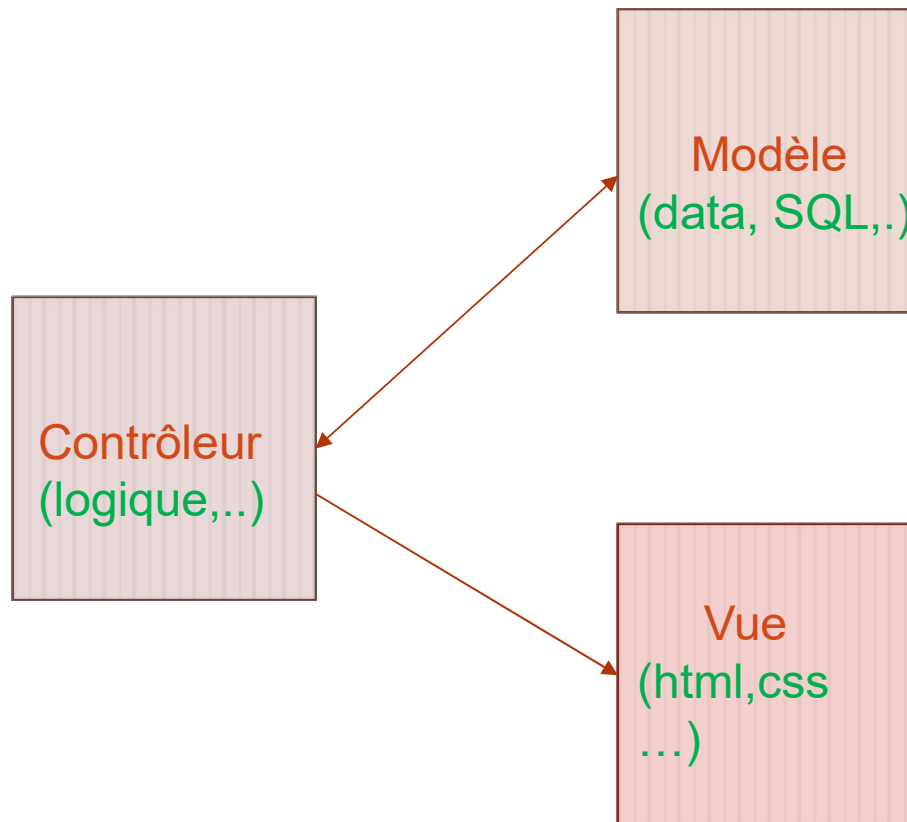
# Modèle MVC

- Ce modèle se compose de **trois** parties : **Modèle**, **Vue** et **Contrôleur**
- **Modèle (Model)** gère les **données** et les opérations qui y sont associées. Il peut s'agir d'une *base de données*, d'un *système de fichiers*, d'un *service web* ou de toute autre source de données. On y trouve des **requêtes SQL**
- **Vue (View)** est responsable de **l'affichage** des données au format HTML, XML ou autre. Elle n'a pas de connaissance directe de la source de données. On y trouve du **HTML** et un peu de **PHP**

- **Contrôleur (Controller)** : gère les **interactions** de l'utilisateur avec l'application. Il sert **d'intermédiaire** entre le modèle et la vue. Il gère la **logique** du code et prend des **décisions**. Il *reçoit* les entrées de l'utilisateur, *effectue* des opérations sur le modèle et *renvoie* une réponse à la vue.



**Fig.** Architecture MVC



### Processus d'échange d'information

- 1.** Le contrôleur demande les données au modèle
- 2.** Le modèle traduit la demande en une requête SQL,  
- Récupère les données et les renvoie au contrôleur
- 3.** Le contrôleur transmet les données récupérées à la vue
- 4.** La vue affiche ces données

## Exemple site e-commerce

- **Modèle**: une **base de données** contenant les produits, les clients et les commandes, ainsi qu'un ensemble de fonctions pour gérer les interactions avec ces données
- **Vue**: les pages **HTML** générées dynamiquement à partir des données du modèle. Par exemple, une page qui affiche la liste des produits avec leur nom, leur prix et un bouton "Commander"
- **Contrôleur**: Par exemple, lorsqu'un utilisateur clique sur le bouton "Commander ", le contrôleur reçoit la demande, ajoute le produit au panier du client dans le modèle, puis renvoie une réponse à la vue pour afficher la mise à jour du panier

Merci pour votre attention