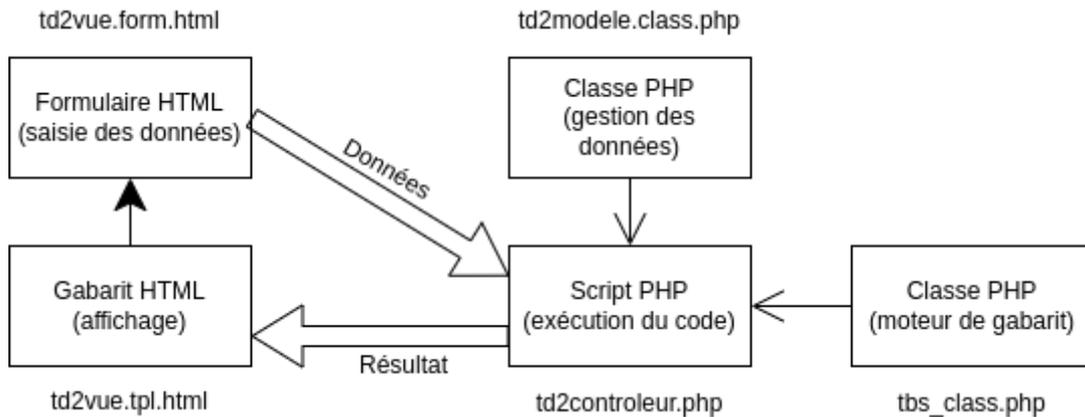


The Big Picture

L'objectif du TD est de transmettre des données saisies sur un formulaire (page td2vue.form.html) à un script PHP (td2controleur.php) qui exécute le code de l'application via une classe PHP (td2modele.class.php) qui gère les données et une bibliothèque (tbs_class.php) qui est un moteur de gabarits. Le code « calcule » un résultat et l'envoie à un gabarit HTML (td2vue.tpl.html) qui affiche le résultat, puis retourne au formulaire de saisie.



Création du formulaire

- Afficher la page <https://eric.univ-lyon2.fr/jdarmont/docs/web/squelette.html>, puis son code HTML (clic droit, code source). Copier/coller le code entier dans un fichier et le sauvegarder sous le nom td2vue.form.html.
- Dans la page td2vue.form.html, créer un formulaire selon les spécifications suivantes (T65-T74).
 - La cible du formulaire est la page td2controleur.php qui sera créée ultérieurement.
 - La méthode de transmission des données est « post ».
 - Le mode d'encodage est « multipart/form-data ».
 - Tous les champs du formulaire sont regroupés dans un *fieldset* de légende « Location de véhicule ».
 - Chaque champ doit faire l'objet d'une description *label* (accessibilité).
 - L'apparence du formulaire et la liste des champs à prévoir sont indiquées ci-contre et ci-dessous, respectivement.

Location de véhicule

Type

Energie

- Essence
- **Electrique**
- GPL

Nombre de jours

Assurance

ID champ	Nom champ	Intitulé	Type	Particularités
type	type	Type	select	Valeurs : Touristique, Utilitaire
essence	energie	Essence	radio	Valeur : Essence
electrique	energie	Electrique	radio	Valeur : Electrique, coché par défaut
gpl	energie	GPL	radio	Valeur : GPL
nbg	nbg	Nombre de jours	text	Taille : 5
assurance	assurance	Assurance	checkbox	
reset			reset	Valeur : Réinitialiser
submit			submit	Valeur : Valider

- Téléverser td2vue.form.html sur phpetu.univ-lyon2.fr. Vérifier l'aspect du formulaire et la validité du code HTML (<https://validator.w3.org>).

Gabarit

1. Copier-coller le code HTML de la page <https://eric.univ-lyon2.fr/jdarmont/docs/web/squelette.html> (clic droit, code source) dans un fichier et le sauvegarder sous le nom td2vue.tpl.html.
2. Y créer un tableau HTML contenant deux cellules <td></td> dont les paramètres sont **champ** et **valeur** (T55) pour afficher les couples (nom de champ, valeur) des données du formulaire.
3. Sous le tableau HTML, ajouter un lien de retour au fichier td2vue.form.html.

Transmission de données et affichage du résultat

FormData
\$post_data : Array
__construct() retournePost()

- 1 Créer un fichier td2modele.class.php. Y placer les balises PHP, puis créer une classe FormData destinée à recueillir et traiter les données venant du formulaire de la page td2vue.form.html, contenant un attribut privé \$post_data, un constructeur qui affecte à \$this->post_data le paramètre \$post_param et une méthode publique retournePost() qui renvoie \$this->post_data.
- 2 Créer le fichier td2controleur.php. Inclure les fichiers tbs_class.php et td2modele.class.php et créer un objet TBS, ainsi qu'un objet \$data de classe FormData avec comme paramètre \$_POST.
- 3 \$_POST, et donc la méthode retournePost(), renvoient un tableau associatif combinant chaque champ du formulaire à sa valeur. À l'aide d'un foreach de type clé/valeur (T44) sur \$data->retournePost(), créer deux tableaux \$tabChamps et \$tabValeurs pour stocker les résultats.
- 4 Mettre en œuvre l'affichage du gabarit pour \$tabChamps et \$tabValeurs (T56).
- 5 Téléverser les fichiers td2vue.tpl.html, td2modele.class.php et td2controleur.php sur le serveur phpetu. Dans le formulaire de la page td2vue.form.html, saisir des données et valider. Vérifier que le résultat est conforme à l'exemple fourni en fin de sujet. Noter la valeur du champ « assurance » en fonction de si l'option est cochée ou non.
- 6 Vérifier la validité du code HTML généré dans td2controleur.php. Bravo, vous venez de coder votre première application web MVC !

Téléversement d'un fichier binaire

FormData
\$post_data : Array \$file_data : Array
__construct() retournePost() retourneFile() uploadePhoto()

- 1 Ajouter au formulaire de la page td2vue.form.html, le champ décrit ci-dessous.

ID champ	Nom champ	Intitulé	Type	Particularités
photo	photo	Photo du véhicule	File	

- 2 Dans le fichier td2vue.tpl.html, créer un nouveau tableau HTML sous le précédent et identique au précédent, à l'exception du nom des paramètres qui sont **champlmg** et **valeurlmg**.
- 3 Télécharger sur le Web une image quelconque (de taille raisonnable) et la sauvegarder sur le bureau.

- 4 Dans le fichier td2modele.class.php, ajouter à la classe FormData un attribut privé \$file_data, son initialisation au paramètre \$file_param dans le constructeur et une méthode publique retourneFile() qui renvoie \$this->file_data.
- 5 Dans la page td2controleur.php, à la création de l'objet \$data, ajouter le paramètre \$_FILES["photo"] en 2^e position.
- 6 \$_FILE, et donc la méthode retourneFile(), renvoient un tableau associatif combinant chaque champ photo à sa valeur. À l'aide d'un foreach de type clé/valeur sur \$data->retourneFile(), créer deux tableaux \$tabChampsImg et \$tabValeursImg pour stocker les résultats. Téléverser les fichiers td2vue.form.html, td2modele.class.php et td2controleur.php sur phpetu. Tester et vérifier que le résultat est conforme à l'exemple fourni en fin de sujet. Il est normal que l'image ne s'affiche pas pour l'instant.
- 7 Créer dans le répertoire de votre site un nouveau répertoire nommé upload situé au même niveau que les pages td2.*.
- 8 Dans le fichier td2vue.tpl.html, ajouter sous le tableau précédent un élément image() dont la source (src) est [onshow.cheminImg].
- 9 Dans le fichier td2modele.class.php, ajouter à la classe FormData une méthode publique uploadePhoto() qui copie le fichier transmis par le formulaire dans le répertoire upload, sous son nom original. Utiliser pour cela la fonction move_uploaded_file(\$source, \$destination). Le chemin du fichier source est le nom temporaire du fichier transféré sur le serveur. Le chemin du fichier de destination est la concaténation de la « constante magique » __DIR__, qui indique le chemin du script PHP en cours d'exécution, du répertoire de destination « /upload/ » et du nom original du fichier transmis.
- 10 Dans la page td2controleur.php, après la préparation des couples champs/valeurs \$tabChampsImg et \$tabValeursImg, appeler la méthode uploadePhoto() pour l'objet \$data.
- 11 Toujours dans td2controleur.php, initialiser le chemin d'accès à la photo, nommé **\$cheminImg**, à la concaténation « ./upload/ » et du nom du fichier transmis, soit \$data->retourneFile()["name"].
- 12 Mettre en œuvre l'affichage du gabarit ppur \$tabChampsImg et \$tabValeursImg.
- 13 Téléverser de nouveau td2vue.tpl.html, td2modele.class.php et td2controleur.php sur phpetu. Dans le formulaire de la page td2vue.form.html, saisir des données (photo du bureau comprise) et valider. Vérifier que le résultat est conforme à l'exemple fourni en fin de sujet.
- 14 Vérifier de nouveau la validité du code HTML de la page td2controleur.php.

Exemple de résultat attendu

Données du
formulaire

Champ	Valeur
type	Utilitaire
energie	Electrique
nbg	2
assurance	on

Données de l'image

Champ	Valeur
name	jb007.png
full_path	jb007.png
type	image/png
tmp_name	/tmp/php648zkb
error	0
size	8940



[Retour au formulaire](#)