



Nom & Prénom :

Order Pizza

Personal Information

Name*
Your name

Phone*
Your phone

Date of birth
jj / mm / aaaa

Page 1 of 6

Next >

Order Summary

Name: Invalid name.
Phone: Invalid phone.
Date of birth: Invalid date.
Pizza size: Select size.
Pizza crust: Regular Crust
Cheese: Mozzarella
Sauce: Pizza Sauce
Toppings: Mushrooms

Total Price = \$1.50

Pizza Pep's est devenue la nouvelle coqueluche du quartier grâce à ses pizzas qui remportent un franc succès et font le bonheur des papilles des habitants. Actuellement, les commandes de cette pizzeria peuvent être passées soit par téléphone, soit directement sur place.

Pour améliorer son service et toucher une audience plus large, le propriétaire a ajouté une page web conviviale permettant de passer des commandes en toute simplicité, sans nécessiter de déplacement ou d'attente interminable au téléphone.

La page web a été développée par un jeune fan de Pizza Pep's. Hélas, il n'est pas très doué en JavaScript, et il sollicite votre aide.

Travail demandé

1. Télécharger les ressources relatives à ce projet.
2. Décompresser l'archive dans un dossier portant votre nom sur le bureau. L'archive contient trois fichiers :
 - **order.html** : code HTML de la page (**ne pas modifier**)
 - **order.css** : code CSS (**ne pas modifier**)
 - **order.js** : code JavaScript (**à compléter**)
3. Ouvrir le fichier **order.js**, puis implémenter les fonctions suivantes :
 - a) **calcPrice()** : retourne le prix du pizza selon ses dimensions et sa garniture.

Pizza Size				Toppings
Personal	Medium	Large	X-Large	Each
\$12	\$15	\$22	\$25	\$1.5

- b) **isValidName(name)** : retourne **true** si la chaîne **name** est composée d'au moins trois lettres alphabétiques et/ou des espaces,
- c) **isValidPhone(phone)** : retourne **true** si la chaîne **phone** contient un numéro de téléphone composé de 8 chiffres, le numéro de téléphone doit obligatoirement commencer par l'un des chiffres suivants : {2, 4, 3, 5, 7, 9},
- d) **updateSummary()** : met-à-jour les informations affichées dans la section **Order Summary** à partir du formulaire.
- e) **orderPizzaClicked(ev)** : vérifie la validité des données entrées par l'utilisateur :
- Le nom est alphabétique et contient au moins 3 caractères,
 - Le téléphone est valide,
 - L'utilisateur a choisi la taille de son pizza,

Order Summary

Name: Invalid name. ← spanName

Phone: Invalid phone. ← spanPhone

Date of birth: Invalid date. ← spanDOB

Pizza size: Select size. ← spanPizzaSize

Pizza crust: Regular Crust ← spanPizzaCrust

Cheese: Mozzarella ← spanPizzaCheese

Sauce: Pizza Sauce ← spanPizzaSauce

Toppings: Mushrooms ← ulTopping

Total Price = \$1.50

Utiliser les fonctions **isValidName()** et **isValidPhone()**.

Si le formulaire n'est pas bien rempli ou qu'une information manque afficher une boîte d'alerte avec le message suivant : "There are invalid inputs. Correct them before continuing."

Si tout va bien une boîte d'alerte affiche : "Order received successfully. You'll be delivered soon, keep your phone near by."

- f) **resetOrderClicked(ev)** : vide le contenu du formulaire, affiche la première page du formulaire en utilisant la fonction **displayPage(numPage)**, appelle les fonctions **updateSummary()** et **updatePrice()**,
4. Attacher la fonction **orderPizzaClicked(ev)** à l'évènement « **submit** » du formulaire.
5. Attacher la fonction **orderPizzaClicked()** à l'évènement « **reset** » du formulaire.

Barème

3.a	3.b	3.c	3.d	3.e	3.f	4	5	Note
3.5	1.75	1.75	4	5.5	1.5	1	1	