

Examen régional de fin de module Année de Formation 2023/2024			
Code module : M204 DEVELOPPEMENT FRONT END			Type d'Epreuve : Pratique
ière	: DDOWFS	Durée	: 2h30mn
nnée	: 2 ° A	Barème global	: / 40

### Exercice 1 : (16 points)

## Ajouter Un livre

Num :

Titre :

Date d'édition :

Genre littéraire :

Prix :



Ajouter

Réinitialiser

- Num : 1 | Titre : Antigone | Date édition : 1944-01-05 | Genre : Tragédie | Prix : 59
- Num : 2 | Titre : Les pauvres | Date édition : 2000-12-30 | Genre : Tragédie | Prix : 109
- Num : 3 | Titre : TED | Date édition : 2010-12-30 | Genre : Comédie | Prix : 80

On souhaite créer un composant <AjoutLivre /> qui permet d'ajouter un Livre à une liste avec un contrôle :

- Tous les champs doivent être renseigné
- Le prix du livre ne doit pas dépasser 200 dh

- 1- Créer le Formulaire avec les contrôles demandés (les messages d'erreur s'affichent dans une alert) (6 pts)
- 2- Créer la fonction AjouterLivre() qui permet d'ajouter un livre à une liste à chaque clique du bouton ajouter (4 pts)
- 3- Afficher la liste des livres en dessous du formulaire (4 pts)
- 4- Créer la fonction Reset qui permet de réinitialiser la liste (2 pts)



**Exercice 2 : (16 points)**

L'objectif est la réalisation d'une application permettant d'afficher l'interface d'une boutique à partir d'un objet JSON :



PC Portable Gamer

2190 DH

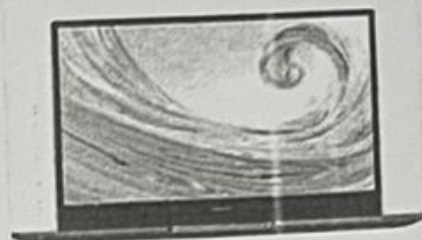
Ajouter au panier



Pc Portable Chromebook Acer

3640 DH

Ajouter au panier



PC Portable - HUAWEI

1270 DH

Ajouter au panier

L'objet JSON :

```
const produits = [
  {
    id: 2,
    title: 'PC Portable Gamer',
    price: '2190 DH',
    thumbnail: 'HP14424U3EA.jpg'
  },
  {
    id: 3,
    title: 'Pc Portable Chromebook Acer',
    price: '3640 DH',
    thumbnail: 'NXATHEF002.jpg'
  },
  {
    id: 4,
    title: 'PC Portable - HUAWEI',
    price: '1270 DH',
    thumbnail: 'HUA6901443442959.jpg'
  }
];
```



Réaliser l'application demandée avec les éléments :

- 1- Déclaration une variable d'état local initialiser par l'objet JSON donné. (4 pts)
- 2- Création d'une composante produit qui représente un pc. (6 pts)
- 3- Réalisation d'une composante permettant d'afficher les produits en format carte. (6 pts)



### Exercice 3 : (8 points)

On dispose d'un Rest-API <https://disease.sh/v3/covid-19/countries> qui renvoie les données covid en format JSON

- 1- Donner un code permettant de récupérer les données de l'API dans une variable d'état local data. (4 pts)
- 2- Donner le code permettant d'afficher les pays et le nombre de cas de covid comme : (4 pts)

```
{
  "country": "nom pays",
  "countryInfo": {
    "_id": nombre,
    "iso2": "Abbrév",
    "iso3": "Abbrév",
    "lat": nombre,
    "long": nombre,
    "flag": "path"
  },
  "cases": nombre any,
  "todayCases": nombre,
  "deaths": nombre,
  "todayDeaths": nombre,
  "recovered": nombre,
  "todayRecovered": nombre,
  "active": nombre,
  "critical": nombre,
  "casesPerOneMillion": nombre,
  "deathsPerOneMillion": nombre,
  "tests": nombre,
  "testsPerOneMillion": nombre,
}
```

Objet de données

- Afghanistan ----- 207748
- Albania ----- 333818
- Algeria ----- 271236
- Andorra ----- 47751
- Angola ----- 105095
- Anguilla ----- 3904
- Antigua and Barbuda ----- 9106
- Argentina ----- 9963697
- Armenia ----- 445976
- Aruba ----- 43752
- Australia ----- 11136734
- Austria ----- 5716267
- Azerbaijan ----- 826496
- Bahamas ----- 37491
- Bahrain ----- 698952
- Bangladesh ----- 2037230

Résultat



Examen régional de fin de module			
Année de Formation 2023/2024			
Code module : M204			Type d'Epreuve : Pratique
DEVELOPPEMENT FRONT END			
Filière	: DDOWFS	Durée	: 2h30mn
Année	: 2 <sup>e</sup> A	Barème global	: / 40

Exercice 1 : (16 points)

## Ajouter Un Film

Num : 3

Titre : TED

Date d'édition : 30/12/2010 ☐

Genre littéraire : Comédie



Ajouter

Réinitialiser

- Num : 1| Titre : Antigone| Date édition :1944-01-05| Genre : Tragédie
- Num : 2| Titre : Les pauvres| Date édition :2000-12-30| Genre : Tragédie
- Num : 3| Titre : TED| Date édition :2010-12-30| Genre : Comédie

On souhaite créer un composant <AjoutFilm /> qui permet d'ajouter un Livre à une liste avec un contrôle :

- Tous les champs doivent être renseigné
  - La date d'édition doit être située entre le « 01/01/1945 » et la date actuelle.
- 1- Créer le Formulaire avec les contrôles demandés (les messages d'erreur s'affichent dans une alert) (6 pts)
  - 2- Créer la fonction AjouterFilm() qui permet d'ajouter un Film à une liste à chaque clique du bouton ajouter (4 pts)
  - 3- Afficher la liste des films en dessous du formulaire (4 pts)
  - 4- Créer la fonction Reset qui permet de réinitialiser la liste (2 pts)





## Exercice 2 : (16 points)

Soit la base de données suivante :

```
[
  {
    nom: "nom1",
    prenom: "prenom1",
    dateNaissance: "15/02/2000",
    ville: "Oujda",
    genre: "F",
    notes : [15,14,12,10]
  },
  .....
]
```

1. Créer une fonction qui permet de calculer et retourner la moyenne des notes (3pts)
2. Créer le composant fonctionnel **<Etudiant />** qui permet d'afficher un étudiant (nom, prénom, moyenne) sous forme d'une ligne du tableau html (5pts)
3. Créer le composant **"HeaderTable"** qui permet d'afficher la ligne de titres du tableau (2pts)

Nom	Prénom	Moyenne
-----	--------	---------

4. En utilisant les deux composants **"Etudiant"** et **"HeaderTable"**, créer le composant fonctionnel **<ListeEtudiants />** qui permet d'afficher une liste des étudiants, passée dans props, sous forme d'un tableau html (3pts)
5. En utilisant le composant **"ListeEtudiants"** et la base de données :
  - Afficher tous les étudiants (1pts)
  - Afficher les étudiants masculins (2pts)



مكتب التكوين المهني وإعانة الشغل  
Office de la Formation Professionnelle  
et de la Promotion du Travail

### Exercice 3 : (8 points)

1. Créer le code JSX qui affiche le formulaire de connexion suivant (2pts)
2. Créer la fonction qui affiche le message d'erreur associé au nom du champ. (2pts)
3. Créer la fonction qui gère la soumission du formulaire (2pts)
4. Afficher le message de réussite après l'envoi : (2pts)  
Je suis {name} né le {date\_naissance} à {ville}  
et mes loisirs sont : {loisirs}

## Inscription

L'identifiant

Mot de passe

Date de naissance

Ville

Genre ☐ Homme ☐ Femme

Loisirs ☐ Sport ☐ Lecture ☐ Musique

Photo

S'inscrire





Examen régional de fin de module  
 Année de Formation 2023/2024

 Code module : M204  
 Développement front-end

 Type d'Epreuve :  
 Pratique

Filière	: Développement Digital option Web Full Stack	Durée	: 2h30mn
Année	: 2 <sup>ème</sup> année	Barème global	: / 40

**Application en react redux :**
**Création d'une application react redux qui permet de gérer les produits :**

 Le but de cette application est de réaliser une application en redux où le store redux permettant de gérer l'état global de l'application : **Gestion des produits**

Le store a un état initial contenant un tableau avec les données sur les produits :

Code	: code de produit,
libelle	: le nom de produit,
prix	: le prix de produit,
Quantite	: la quantité de produit,
DateP	: la date de production de produit,
DateE	: la date d'expiration de produit

Le reducer permettant l'accès aux données, l'ajout, la modification et la suppression d'un produit.

**Travail à faire:**
**1. Créer le store redux décrit ci-dessus: (8 pts)**

- Créer le fichier Action.js qui comporte 3 actions. (1.5pts)
- Créer le fichier Store.js (1pt)
- Importer le store dans le provider de votre fichier index de projet react (1.5pts)
- Créer le fichier Reducer.js (4 points)


**2. Créer un composant « Index » pour accéder aux différents composants: (4 pts)**
**3. créer un composant Q3 qui permet d'afficher la liste des produits dans un tableau avec les boutons « Modifier » et « Supprimer » : (7 pts)**

- Récupérer le tableau des produits (4pts)
- Le clic sur le bouton modifier nous ramène vers le composant Q4 en passant le code de produit en paramètres de lien. (1pts)
- Le clic sur le bouton Supprimer nous ramène vers une fonction qui permet d'envoyer le code de produit dans le dispatch afin de faire la suppression de produit concerné. (2pts).

Q3 Q4 Q5 Q6

**Liste des produits**

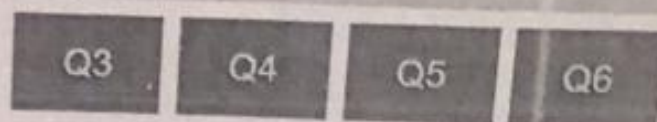
localhost:3000/Q4/1

Code	Libelle	Prix	Quantite	Date de production	Date d'expiration	Actions	
1	prd1	120	5	12/01/2020	12/01/2022	Modifier	Supprimer
2	prd2	220	50	12/02/2021	12/07/2023	Modifier	Supprimer
3	prd3	320	15	12/03/2020	12/08/2026	Modifier	Supprimer
4	prd4	420	105	12/04/2022	12/09/2025	Modifier	Supprimer



4. Créer un composant Q4 qui permet de modifier une voiture: (10 pts)

- La valeur de l'ID est récupérée depuis la barre d'adresse. (1pt)
- Déclarez trois états (useState) : la marque, le modèle et le type où chaque état prend la valeur par défaut convenable. (3pts)
- Dans la partie "return" de la fonction créer le formulaire jsx qui est organisé dans un tableau où les valeurs par défaut des champs sont récupérées en se basant sur l'ID de voiture (le champs ID est désactivé en modification). (4pts)
- Créer la fonction qui permet d'envoyer les nouvelles valeurs dans le dispatch. (2pts)



## Liste des voitures

ID : 1

Marque : BMW

Modèle : 2013

Type : essence

**Enregistrer**

5. Créer un composant Q5 qui permet d'ajouter une voiture comme il est indiqué dans le formulaire ci-dessous : (7 pts)

- Déclarez quatre états (useState) pour l'id, la marque, le modèle et le type. (2pts)
- Dans la partie "return" de la fonction créer le formulaire jsx qui est organisé dans un tableau. (3pts)
- Créer la fonction qui permet d'envoyer les valeurs de nouvelle voiture dans le dispatch. (2pts)



## Ajouter une voiture

ID : ID

Marque : Marque

Modèle : Modèle

Type : Type

**Enregistrer**



6. Créer un composant Q6 qui permet d'afficher la liste des voitures qui ont une ancienneté plus que 5 ans : (4pts)



## Liste des voitures qui ont une ancienneté plus que 5 ans

ID	Marque	Modèle	Type	Actions	
1	BMW	2013	diesel	Modifier	Supprimer
3	AUDIT	2010	essence	Modifier	Supprimer
4	ALPHA ROMIO	2015	essence	Modifier	Supprimer



Examen régional de fin de module  
Année de Formation 2023/2024

Code module : M204  
Développement front-end

Type d'Epreuve :  
Pratique

Filière	: Développement Digital option Web Full Stack	Durée	: 2h30mn
Année	: 2 <sup>ème</sup> année	Barème global	: / 40

Application en react redux :

Création d'une application react redux qui permet de gérer les utilisateurs:

Le but de cette application est de réaliser une application en redux où le store redux permettant de gérer l'état global de l'application : **gestion des utilisateurs**

Le store a un état initial contenant un tableau avec les données sur les utilisateurs :

id : Id de l'utilisateur,  
nom : le nom de l'utilisateur,  
age : L'âge de l'utilisateur,  
email : L'email de l'utilisateur

Le reducer permettant l'accès aux données, l'ajout, la modification et la suppression d'un utilisateur.

Travail à faire:

1. Créer le store redux décrit ci-dessus: (8 pts)

- Créer le fichier Action.js qui comporte 3 actions. (1.5pts)
- Créer le fichier Store.js (1pt)
- Importer le store dans le provider de votre fichier index de projet react (1.5pts)
- Créer le fichier Reducer.js (4 points)



2. Créer un composant « Index » pour accéder aux différents composants: (4 pts)

3. Créer un composant Q3 qui permet d'afficher la liste des utilisateurs dans un tableau avec les boutons « Modifier » et « Supprimer »: (7 pts)

- Récupérer le tableau des utilisateurs (4pts)
- Le clic sur le bouton modifier nous ramène vers le composant Q4 en passant l'id de l'utilisateur en paramètres de lien. (1pts)
- Le clic sur le bouton Supprimer nous ramène vers une fonction qui permet d'envoyer l'id de l'utilisateur dans le dispatch afin de faire la suppression de l'utilisateur concerné. (2pts).

Q3 Q4 Q5 Q6

Liste des utilisateurs

localhost:3000/Q4

ID	Nom	Age	Email	Action
1	ahmed karim	17	ahmed.karim@gmail.com	Modifier Supprimer
2	fatima ramdani	22	Fatima.ramdani@gmail.com	Modifier Supprimer
3	said aknoun	16	said.aknoun@gmail.com	Modifier Supprimer
4	karima hassnaoui	26	karima.hassnaoui@gmail.com	Modifier Supprimer



4. Créer un composant Q4 qui permet de modifier un utilisateur: (10 pts)

- La valeur de l'ID est récupérée depuis la barre d'adresse. (1pt)
- Déclarez trois états (useState) : le nom, l'âge et l'email où chaque état prend la valeur par défaut convenable. (3pts)
- Dans la partie "return" de la fonction créer le formulaire jsx qui est organisé dans un tableau où les valeurs par défaut des champs sont récupérées en se basant sur l'ID de l'utilisateur (le champs ID est désactivé en modification). (4pts)
- Créer la fonction qui permet d'envoyer les nouvelles valeurs dans le dispatch. (2pts)

## Modifier un utilisateur

Q3 Q4 Q5 Q6

ID : 1  
Nom : ahmed karim  
Age : 17  
Email : ahmed.karim@gmail.com

Enregistrer

5. Créer un composant Q5 qui permet d'ajouter un utilisateur comme il est indiqué dans le formulaire ci-dessous : (7 pts)

- Déclarez quatre états (useState) pour l'id, le nom, l'âge et l'email. (2pts)
- Dans la partie "return" de la fonction créer le formulaire jsx qui est organisé dans un tableau. (3pts)
- Créer la fonction qui permet d'envoyer les valeurs de nouveau utilisateur dans le dispatch. (2pts)

## Ajouter un utilisateur

Q3 Q4 Q5 Q6

ID : id  
Nom : Nom  
Age : Age  
Email : Email

Enregistrer



6. Créer un composant Q6 qui permet d'afficher la liste des utilisateurs qui sont mineurs (age < 18ans): (4pts)

## Liste des utilisateurs qui sont mineurs

ID	Nom	Age	Email	Actions	
1	ahmed karim	17	ahmed.karim@gmail.com	Modifier	Supprimer
3	said aknoun	16	said.aknoun@gmail.com	Modifier	Supprimer



## Examen régional de fin de module

Année de Formation 2023/2024

Code module : M204  
Développement front-endType d'Epreuve :  
Pratique

Filière	Développement Digital option Web Full Stack	Durée	2h30mn
Année	2 <sup>ème</sup> année	Barème global	/ 40

Application en react redux :

Création d'une application react redux qui permet de gérer les voitures:

Le but de cette application est de réaliser une application en redux où le store redux permettant de gérer l'état global de l'application : **gestion des voitures**

Le store a un état initial contenant un tableau avec les données sur les voitures :

id : Id de voiture,  
 marque : La marque de voiture,  
 modele : Le modèle de la voiture,  
 type : Type essence ou diesel

Le reducer permettant l'accès aux données, l'ajout, la modification et la suppression d'une voiture.

Travail à faire:

## 1. Créer le store redux décrit ci-dessus: (8 pts)

- Créer le fichier Action.js qui comporte 3 actions. (1.5pts)
- Créer le fichier Store.js (1pt)
- Importer le store dans le provider de votre fichier index de projet react (1.5pts)
- Créer le fichier Reducer.js (4 points)



## 2. Créer un composant « Index » pour accéder aux différents composants: (4 pts)

## 3. Créer un composant Q3 qui permet d'afficher la liste des utilisateurs dans un tableau avec les boutons « Modifier » et « Supprimer »: (7 pts)

- Récupérer le tableau des voitures (4pts)
- Le clic sur le bouton modifier nous ramène vers le composant Q4 en passant l'id de voiture en paramètres de lien. (1pts)
- Le clic sur le bouton Supprimer nous ramène vers une fonction qui permet d'envoyer l'id de voiture dans le dispatch afin de faire la suppression de voiture concerné. (2pts).

Q3 Q4 Q5 Q6

## Liste des voitures

localhost:3000/Q4/

ID	Marque	Modèle	Type	Actions	
1	BMW	2013	diesel	Modifier	Supprimer
2	GOLF 8	2018	diesel	Modifier	Supprimer
3	AUDIT	2010	essence	Modifier	Supprimer
4	ALPHA ROMEO	2015	essence	Modifier	Supprimer

4. Créer un composant Q4 qui permet de modifier une voiture (10 pts)

- La valeur de l'ID est récupérée depuis la barre d'adresse. (1pt)
- Déclarez trois états (useState) : la marque, le modèle et le type où chaque état prend la valeur par défaut convenable. (3pts)
- Dans la partie "return" de la fonction créer le formulaire jsx qui est organisé dans un tableau où les valeurs par défaut des champs sont récupérées en se basant sur l'ID de voiture (le champs ID est désactivé en modification). (4pts)
- Créer la fonction qui permet d'envoyer les nouvelles valeurs dans le dispatch. (2pts)

Q3 Q4 Q5 Q6

### Liste des voitures

ID: 3

Marque: BMW

Modèle: 2013

Type: essence

Enregistrer

5. Créer un composant Q5 qui permet d'ajouter une voiture comme il est indiqué dans le formulaire ci-dessous : (7 pts)

- Déclarez quatre états (useState) pour l'id, la marque, le modèle et le type. (2pts)
- Dans la partie "return" de la fonction créer le formulaire jsx qui est organisé dans un tableau. (3pts)
- Créer la fonction qui permet d'envoyer les valeurs de nouvelle voiture dans le dispatch. (2pts)

Q3 Q4 Q5 Q6

### Ajouter une voiture

ID: 1

Marque: BMW

Modèle: 2013

Type: essence

Enregistrer

6. Créer un composant Q6 qui permet d'afficher la liste des voitures qui ont une ancienneté plus que 5 ans : (4pts)

Q3 Q4 Q5 Q6

### Liste des voitures qui ont une ancienneté plus que 5 ans

ID	Marque	Modèle	Type	Actions	
1	BMW	2013	diesel	Modifier	Supprimer
3	AUDIT	2010	essence	Modifier	Supprimer
4	ALPHA ROMIO	2015	essence	Modifier	Supprimer

