



# SpringBoot Exercice

ABDERRAHMAN CHEIKH  
AHMEDOU

SOFTWARE ARCHITECT AT  
GTI

# Exercice 1

---

- Créer un controller avec méthode qui retourne 'Hello world'
- Utiliser les annotation : `@Controller` et `@RequestMapping`, `@ResponseBody`

# Exercice 2

---

- Changer le port de l'application, mettre 9090

# Exercice 3

---

- Refaire l'exo 1 en utilisant `@RestController` et `@RequestMapping`

# Exercice 4

---

- Calculer la somme de 2 nombres que vous envoyez via Get

Ex : <http://localhost:9090/exo4?nombre1=2&nombre2=3>

# Exercice 5

---

- Refaire l'exo 4 mais rendre nombre2 optionnel et si n'est pas envoyé le résultat sera nombre1
- Refaire la même chose en donnant une valeur par défaut à nombre1

# Exercice 6

---

- Retourner le carré du nombre dans l'url. Ex. 4 => 16

<http://localhost:9090/exo6/4>

# Exercice 7

---

- Ecrire une méthode qui permet de recevoir paramètres via GET sans utiliser `@RequestParam` ni `PathVariable` :
- login: abdou
- mdp : Test



# Exercice 8

---

- Ecrire une méthode qui permet de recevoir paramètres via GET sans utiliser `@RequestParam` ni `PathVariable` et la méthode a seulement 3 paramètres:
- login: abdou
- mdp : Test
- tel: 42004204
- email: a.sidielmoctar@gamil

# Exercice 9

---

- Ecrire une méthode qui permet de recevoir via Post l'objet suivant:
- login: abdou
- mdp : Test
- tel: 42004204
- email: [a.sidielmoctar@gamil](mailto:a.sidielmoctar@gamil)

Utiliser : @RestController et @RequestBody

Faire sans utiliser @RequestBody

NB: Utiliser Map<String, Object> ou un DTO

# Exercice 10

---

- UserDto (id, nom, tel, email, password)
- Créer une liste de UserDto avec données
- Récupérer la liste de user
- Récupérer user par id
- Ajouter la possibilité de modifier un user et retourner la liste
- Supprimer un user by id

# Exercice 11

---

Créer une base de données « iscae\_sb\_db »

Créer deux tables :

- genre (id, code)
- personne (id, prenom, nom, tel, email, id\_genre)

Les id sont PK auto-increment

# Exercice 12

---

Il faut avoir les dépendences :

spring-boot-starter-test

spring-boot-starter-data-rest

spring-boot-starter-data-jpa

spring-boot-starter-web

mysql-connector-j

# Exercice 13

---

Ajouter `@EnableJpaRepositories` pour supporter les repositories

Créer `PersonneRepository`

Retourner la liste des personnes

Retourner une personne par id

Retourner une personne par nom

Retourner une personne par tel

# Exercice 14

---

Ajouter une personne

Modifier une personne

Supprimer une personne