

Mongoose & Node.js

◆ 1. Connexion à MongoDB avec Mongoose

```
mongoose.connect(process.env.MONGO_URI, {  
  useNewUrlParser: true,  
  useUnifiedTopology: true  
});
```

L'URI est stocké dans `.env` :

```
MONGO_URI="your-mongodb-uri"
```

◆ 2. Définir le modèle Person

```
const personSchema = new mongoose.Schema({  
  name: { type: String, required: true },  
  age: Number,  
  favoriteFoods: [String]  
});
```

◆ 3. Créer et sauvegarder une personne

```
const person = new Person({  
  name: 'John',  
  age: 30,  
  favoriteFoods: ['pizza', 'tacos']  
});  
  
await person.save();
```

◆ 4. Créer plusieurs personnes

```
const arrayOfPeople = [  
  { name: 'Mary', age: 25, favoriteFoods: ['salad'] },  
  { name: 'Paul', age: 28, favoriteFoods: ['steak'] },  
  { name: 'John', age: 22, favoriteFoods: ['sushi'] }  
];  
  
await Person.create(arrayOfPeople);
```

◆ 5. Trouver toutes les personnes avec `.find()`

```
const people = await Person.find({ name: 'John' });
```

◆ 6. Trouver une personne avec `.findOne()`

```
const person = await Person.findOne({ favoriteFoods: 'tacos' });
```

◆ 7. Trouver une personne par son `_id`

```
const person = await Person.findById(personId);
```

◆ 8. Modifier puis sauvegarder une personne

```
const person = await Person.findById(personId);  
person.favoriteFoods.push('hamburger');  
await person.save();
```

◆ 9. Mettre à jour avec `.findOneAndUpdate()`

```
const updatedPerson = await Person.findOneAndUpdate(  
  { name: personName },  
  { age: 20 },  
  { new: true }  
);
```

◆ 10. Supprimer une personne par **`_id`**

```
const removed = await Person.findByIdAndRemove(personId);
```

◆ 11. Supprimer plusieurs personnes

```
const result = await Person.deleteMany({ name: 'Mary' });
```

◆ 12. Requête enchaînée avec filtres

```
const people = await Person.find({ favoriteFoods: 'burritos' })  
  .sort({ name: 1 })  
  .limit(2)  
  .select('name favoriteFoods')  
  .exec();
```

💡 **Note :** Pour utiliser `await`, assurez-vous d'être dans une fonction `async`.