

Notions

- ▶ **Événements** : ils permettent aux composants de réaliser des traitements et de modifier les différentes valeurs des « states » par l'intermédiaire de la fonction « `setState` ».
- ▶ Les événements sont en général créés au niveau d'un « container », utilisent des fonctions de classes et s'écrivent selon deux conventions :
 - ▶ xxxxx**Handler** ou **handle**xxxxx
- ▶ Pour transmettre une fonction à un composant fils, il suffit de lui transmettre la **référence** par l'intermédiaire d'une propriété
- ▶ Pour ajouter des paramètres de fonctions lors de la transmission au composant fils, il faudra passer par la fonction « **bind** », ou utiliser le système des **fonctions fléchées**.

Exemples

```
anniversaireHandler = (numPersonne) => {
  const newPersonne = {...this.state.personnes[numPersonne]}; //génère une copie de la personne sur laquelle on a cliquée
  newPersonne.age++; //augmente l'age de la personne copiée

  const newTab = [...this.state.personnes]; //on duplique le tableau de personnes
  newTab[numPersonne] = newPersonne; // on remplace la personne à l'indice du tableau sur lequel on a cliqué par la nouvelle personne qu'on a créée
  this.setState({personnes : newTab}); // on remplace dans le state le tableau de personnes par le nouveau tableau
}
```

```
render(){
  return (
    <div>
      <Horloge />
      <Personne
        nom={this.state.personnes[0].nom}
        age={this.state.personnes[0].age}
        save={this.state.personnes[0].save}
        monAnniv={this.anniversaireHandler.bind(this,0)}
      />
      <Personne
        nom={this.state.personnes[1].nom}
        age={this.state.personnes[1].age}
        save={this.state.personnes[1].save}
        monAnniv={this.anniversaireHandler.bind(this,1)}
      />
      <Personne
        nom={this.state.personnes[2].nom}
        age={this.state.personnes[2].age}
        save={this.state.personnes[2].save}
        monAnniv={this.anniversaireHandler.bind(this,2)}
      />
    </div>
  );
}
```

```
render(){
  return (
    <div>
      <Horloge />
      <Personne
        nom={this.state.personnes[0].nom}
        age={this.state.personnes[0].age}
        save={this.state.personnes[0].save}
        monAnniv={() => this.anniversaireHandler(0)}
      />
      <Personne
        nom={this.state.personnes[1].nom}
        age={this.state.personnes[1].age}
        save={this.state.personnes[1].save}
        monAnniv={() => this.anniversaireHandler(1)}
      />
      <Personne
        nom={this.state.personnes[2].nom}
        age={this.state.personnes[2].age}
        save={this.state.personnes[2].save}
        monAnniv={() => this.anniversaireHandler(2)}
      />
    </div>
  );
}
```

```
const Personne = (props) =>{
  return (
    <div>
      <h1 className="monH1">{props.nom}</h1>
      <AgePersonne age= {props.age}/>
      <div>Sexe : {(props.sexe) ? "Homme" : "Femme"}</div>
      <button onClick={props.monAnniv}>Anniversaire</button>
    </div>
  );
}
```