Travel Planner

Arrow Functions

1. **Simple Arrow Function Conversion**

   - **Objective**: Convert a traditional function into an arrow function.
   - **Exercise**: Given a function `getDestinationName` that returns the name of a destination from your travel planner, convert it into an arrow function.
   - **Initial Code**:

   ```
   function getDestinationName(destination) {
     return destination.name;
   }
   ```

2. **Control Flow in Arrow Functions**

   - **Objective**: Understand how to use control flow statements (like if-else) in arrow functions.
   - **Exercise**: Given a function `getTravelStatus` that returns a string stating whether the destination is "Visited", "Upcoming", or "Wishlist" based on a specific status string. Convert this function into an arrow function.
   - **Initial Code**:

   ```
   function getTravelStatus(status) {
     if (status === "Visited") {
       return "You have visited this place";
     } else if (status === "Upcoming") {
       return "You are planning to visit this place";
     } else {
       return "This place is in your wishlist";
     }
   }
   ```

3. **Arrow Functions with Multiple Operations**

- **Objective**: Learn how to perform multiple operations in arrow functions before returning a value.
- **Exercise**: Given a function `getTotalBudget` that takes an array of destinations (each destination has a `budget` property), calculates the total budget of all destinations, and returns the result. Convert this function into an arrow function.
- **Initial Code**:

```javascript
function getTotalBudget(destinations) {
  let totalBudget = 0;
  for (let i = 0; i < destinations.length; i++) {
    totalBudget += destinations[i].budget;
  }
  return 'Total travel budget is ' + totalBudget;
}
```

4. Arrow Functions as Callbacks

- **Objective**: Understand how arrow functions can be used as callbacks.
- **Exercise**: Your travel planner sorts destinations by their budget before displaying. Currently, it uses a traditional function for sorting. Convert this sorting function into an arrow function.
- **Initial Code**:

```javascript
let sortedDestinations = destinations.sort(function(a, b) {
  return b.budget - a.budget;
});
```

5. Implicit Return with Arrow Functions

- **Objective**: Understand the concept of implicit return in arrow functions.
- **Exercise**: There's a function `getDestinationNames` which returns an array of all destination names. Convert this function into a single-line arrow function using the implicit return feature.
- **Initial Code**:

```javascript
function getDestinationNames(destinations) {
  return destinations.map(function(destination) {
    return destination.name;
  });
}
```