

Travel Planner

String methods

For these exercises, you may need the following methods: `toLowerCase()`, `toUpperCase()`, `split()`, `join()`, `includes()`, `replace()`, `parseInt()`, `trim()`, `slice()`, `substring()`, and `charAt()`.

1. **City Name Capitalization:** Write a function that takes a city name in all lowercase and returns it in title case.

Input: `"new york"`

Output: `"New York"`

2. **Flight Search:** Write a function that takes an array of flight destinations and a string search query. The function should return a new array with only the destinations that contain the query string.

Input: `["New York", "Tokyo", "Paris", "London"], "on"`

Output: `["London"]`

3. **Flight Code Format:** Write a function which takes a flight code in the format "code-number" and swaps it to "number-code".

Input: `"JFK-123"`

Output: `"123-JFK"`

4. **Flight Duration:** Write a function that takes a string like "2h 30min" and returns an object with the number of hours and minutes as integers.

Input: `"3h 45min"`

Output: `{ hours: 3, minutes: 45 }`

5. **Travel Agency Acronym:** Write a function that takes a travel agency name and returns its acronym.

Input: `"Global Travel Agency"`

Output: `"GTA"`

6. **Concatenate Travel Details:** Write a function that takes an object with travel details like flight, destination, and date and returns a single string that combines all this information.

Input: `{ flight: "JFK-123", destination: "London", date: "2023-07-16" }`

Output: `"Flight: JFK-123, Destination: London, Date: 2023-07-16"`

7. **Location Slug:** Write a function that takes a location name and returns a URL-friendly slug.

Input: `"San Francisco"`

Output: `"san-francisco"`

8. **Itinerary Trimming:** Write a function that trims the itinerary's details to a specified length without cutting off words.

Input:

`"Visit the Golden Gate Bridge, take a walk in Golden Gate Park, see Alcatraz Island", 50`

Output: `"Visit the Golden Gate Bridge, take a walk in Golden..."`

9. **Travel Wishlist:** Write a function that takes a string of desired destinations separated by commas and converts it into an array of individual destinations.

Input: `"Paris, Tokyo, New York"`

Output: `["Paris", "Tokyo", "New York"]`

10. **Secret Message Decoder:** Write a function that takes an encoded secret message and returns the decoded version. Encode a secret message by replacing all vowels with numbers (e.g., "1" for "a", "2" for "e", etc.).

Input: `"2v2ry tr1v3l3r h1s 4 s3cr3t d3st1n1t10n"`

Output: `"Every traveler has a secret destination"`