



## Introduction to Python Tuples

- Tuples are ordered collections of data items.
- Stored within round brackets ().
- Immutable: Cannot be altered after creation.
- Example: `tuple1 = (1, 2, 3)`





## Tuple Examples

- tuple1 = (1, 2, 2, 3, 5, 4, 6)
- tuple2 = ('Red', 'Green', 'Blue')
- Output: (1, 2, 2, 3, 5, 4, 6)
- Output: ('Red', 'Green', 'Blue')





## Tuple Indexes

- Each tuple item has a unique index.
- Indexes start at 0 (first item).
- Examples: country = ('Spain', 'Italy', 'India')

```
response = requests.get(url)

# checking response.status_code (if you get 502, try rerunning the code)
if response.status_code != 200:
    print(f"Status: {response.status_code} - Try rerunning the code")
else:
    print(f"Status: {response.status_code}\n")

# using BeautifulSoup to parse the response object
soup = BeautifulSoup(response.content, "html.parser")

# finding Post images in the soup
images = soup.find_all("img", attrs={"alt": "Post image"})

# printing images
```





## Accessing Tuple Items: Positive Indexing

- Use positive indexes to access items.
- Example: `country[1]` returns 'Italy'.
- `country[3]` returns 'England'.







## Accessing Tuple Items: Negative Indexing

- Negative indexes start from -1 (last item).
- Example: `country[-1]` returns 'Germany'.
- `country[-3]` returns 'India'.





## Check for Item Presence

- Use 'in' keyword to check if an item exists.
- Example: if 'Germany' in country:  
print('Present')

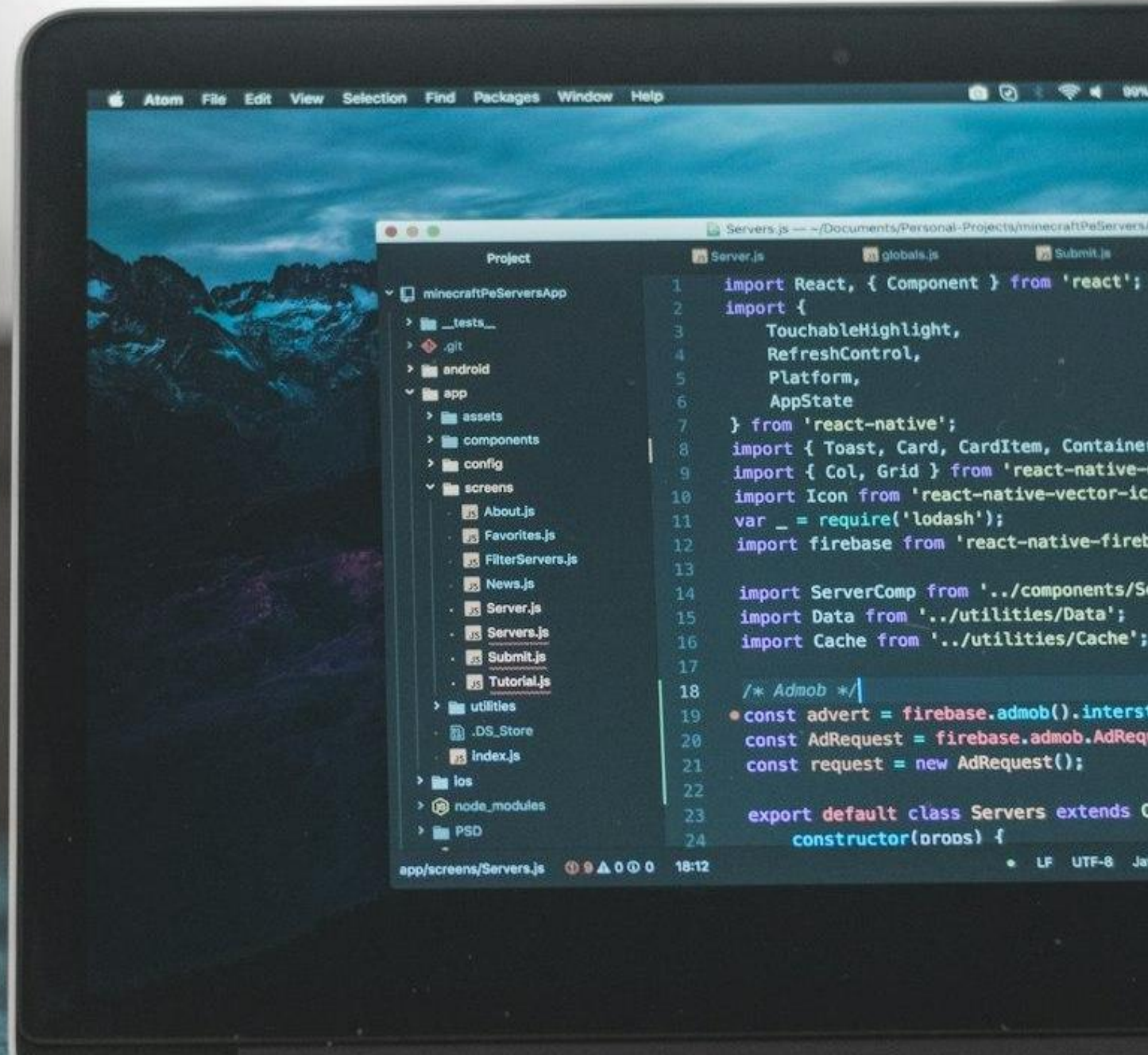






## Range of Index

- Print a range of items using [start:end].
- Optional: Specify a step size (jumpIndex).
- Examples:  
animals[3:7],  
animals[:6].







## Printing Alternate Values

- `animals[::2]` prints alternate values.
- `animals[1:8:3]` prints every 3rd value in range.

