

```
expect(response.data.user.name.first).toBe("Uncle");  
expect(response.data.user.name.last).toBe("Phil");  
expect(response.data.user.role).toBe("parent");
```

```
import * as BelAir from "../bel-air-academy";  
  
●test("uncle phil", async () => {  
  const response = await BelAir.request("philip");  
  
  expect(response.data.user.name.first).toBe("Uncle");  
  expect(response.data.user.name.last).toBe("Phil");  
  expect(response.data.user.role).toBe("parent");  
});
```

# Redundant

```
expect(response.data.user.name.first).toBe("Uncle");  
expect(response.data.user.name.last).toBe("Phil");  
expect(response.data.user.role).toBe("parent");
```

```
expect(response.data.user.name.first).toBe("Uncle");  
expect(response.data.user.name.last).toBe("Phil");  
expect(response.data.user.role).toBe("parent");
```

```
import * as BelAir from "../bel-air-academy";
```

```
●test("uncle phil", async () => {  
  const response = await BelAir.request("philip");  
  
  expect(response.data.user.name.first).toBe("Uncle");  
  expect(response.data.user.name.last).toBe("Phil");  
  expect(response.data.user.role).toBe("parent");  
});
```

```
first).toBe("Uncle");  
last).toBe("Phil");  
toBe("parent");
```

# Redundant

```
expect(response.data.user.name.first  
expect(response.data.user.name.last  
expect(response.data.user.role).to
```

```
expect(response.data.user.name.first).toBe("Uncle");  
expect(response.data.user.name.last).toBe("Phil");  
expect(response.data.user.role).toBe("parent");
```