

# Unit One Tasks: Build a Student Object

---

This task guides you to implement a `Student` object in Sui Move using what you learned in Unit One. You will mint demo and custom `Student` objects and view them in the Sui Explorer.

Path references and lessons are local to this repo to keep everything beginner-friendly.

## Quick Links

- Lessons: `lessons/`
  - `1_set_up_environment.md`
  - `2_sui_project_structure.md`
  - `3_custom_types_and_abilities.md`
  - `4_functions.md`
  - `5_contract_deployment.md`

## Lesson Summary

- Setup: Installs Sui CLI, sets networks, faucet tokens, and basic commands.
- Project Structure: Explains packages, modules, `Move.toml`, addresses, and how to declare `module package::module;`.
- Types & Abilities: Introduces structs and abilities (`copy`, `drop`, `store`, `key`); assets use `has key`, `store` and include an `id: UID`.
- Functions: Shows visibility, return values, `TxContext`, method-call syntax (e.g., `ctx.sender()`), and the mint pattern.
- Deployment: Uses `sui client publish` to deploy and `sui client call` to invoke entry functions; view objects in the explorer.

## Task Specification

PROF

- Define `Student` with fields:
  - `id: UID`
  - `name: string::String`
  - `department: string::String`
  - `faculty: string::String`
- Add abilities: `has key, store`.
- Implement two entry functions:
  - `mint_demo(ctx: &mut TxContext)`: mints a demo student with sample values.
  - `mint_with_values(name, department, faculty, ctx: &mut TxContext)`: mints using caller-supplied values.
- Place your work in: `task_one.move`.

## Hints

- Import strings: `use std::string;`
- Create an object ID: `object::new(ctx)`.

- Send the object to the caller: `transfer::public_transfer(object, ctx.sender())`.
- Use method-call syntax: `ctx.sender()` and string literal conversion: `b"...".to_string()`.
- Entry functions: last parameter is `&mut TxContext` and they return no value.

## Helpful Snippets

- From Hello World object definition:

```
public struct HelloWorldObject has key, store {
    id: UID,
    text: string::String,
}
```

- Method-call syntax examples:

```
let sender = ctx.sender();
let text = b"Hello World!".to_string();
```

## CLI Guide

- Build from task directory:

```
sui move build
```

- Publish (absolute path recommended):

```
sui client publish
```

- Export package ID:

```
export TASK_ONE_PACKAGE_ID=<paste your package object ID>
```

- Call demo mint:

```
sui client call --function mint_demo --module task_one --package
$TASK_ONE_PACKAGE_ID
```

- Call custom mint with typed args:

```
sui client call --function mint_with_values --module task_one --  
package $TASK_ONE_PACKAGE_ID --args "string:Jane Doe"  
"string:Mathematics" "string:Science"
```

## Troubleshooting

- Missing `std::string` import leads to compile errors.
- Objects with `has key, store` must include `id: UID`.
- CLI args must be typed (e.g., `string:...`) and quoted.
- Ensure your Sui client is on the correct network and has gas.

## What to Submit

- The package object ID after publishing.
  - The object ID(s) of minted Student objects.
- github repo of the tasks

## References

- Lessons: [lessons](#)