

Code Conventions

Naming Conventions

Names should always be meaningful and self explanatory, even if it means the name is long - better to have a long name than a confusing one.

This is true for any name in the project - class, member variable, member function, constants, files etc.

Enums

Declaration is on upper camel. Enumerator values are all capital. If there is a space on the name, we use an underline:

```
public enum EnemyState
{
    WALK,
    JUMP,
    OUT_OF_SIGHT
}
```

Constants

All capital. If there is a space on the name, we use an underline:

```
const int MAX_HP = 100;
```

Classes names

Upper camel:

```
public class InputReceiver : MonoBehaviour
```

Public class variables

Lower camel:

```
public GameObject sphere;
```

Protected and private class variables

Lower camel with underling in front:

```
private float _timeCounter;
```

Local variables

Preceded by __:

```
int __currentGold;
```

Methods

Upper camel. Arguments have a p_ in front:

```
public void CalculateDamage(int p_armorPower)
```

Order & Structure

Entities are defined in the following order:

- Classes (in 99% of the cases, one per file)
 - Constants
 - Enums
 - Static member variables
 - Public
 - Protected
 - Private
 - Auto member variables
 - Public
 - Protected
 - Private
 - Functions