

1. design and implement a class for an RPG character

## think about

- how and where to store data
  - what operations are needed
  - how and where to compute information

# how & where to store data

เก็บไว้ใน public class RPGCharactor

၁၃၅

```
private int level; // level ត៉វលេគទ  
private int maxHP; // លំនៅិវិទសុងសុត  
private int maxMana; // មានាសុងសុត  
private double baseRunSpeed;  
// គ្រប់រីយៈពិន្ទុន  
private Integer swordLevel; // level បាប  
private Integer shieldLevel; // level លូ'
```

**what operation are needed**

operation(ฟังก์ชัน) ใน public class RPGCharacter เช่น

```
public void equipSword(int swordLevel) //ใส่ดาบ
public void equipShield(int shieldLevel) //ใส่โล่
public void levelUp() //เพิ่มlevel
public double attack() //ค่าโจมตี
public double defend() //ค่าป้องกัน
private void updateStats()
// อัปเดตHPและMana ตาม level
```

```
public int getMaxHP() //ค่าพลังชีวิตสูงสุด  
public int getMaxMana() //ค่า mana สูงสุด  
public double getBaseRunSpeed()  
//ค่าความเร็วเคลื่อนที่พื้นฐาน  
public int getLevel() //แสดง level
```

1. design and implement a class for an RPG character

think about

- how and where to store data
- what operations are needed
- how and where to compute information

## how & where to compute the information

ใน public class RPGCharacter นำค่าจาก Data มาคำนวณใน Operation เช่น

public void equipShield(int shieldLevel) ได้โลล์ต้องนำ shieldLevel และ baseRunSpeed จาก Data มาคำนวณในการจำลองการได้โลล์ที่ความเร็วจะลดลงเนื่องจากน้ำหนักโลล์ตาม level โลล'

```
// ได้โลล์
public void equipShield(int shieldLevel) { 1 usage  • Korn Sriwattana
    this.shieldLevel = shieldLevel;
    this.baseRunSpeed -= this.baseRunSpeed * (0.1 + 0.08 * shieldLevel);
}
```