Design Document

There are interfaces : Character and Head(Accessories)

Character has 2 types : Archer and Mage

Head has 2 types : Crown and Mask

Class Character

variables

```
protected int level;
protected String name,job;
protected int hp,atk,def;
protected boolean hok,skillok;
protected Head h;
```
hok = use to check helmet is on the head or not, skillok = use to check skill using

construction and methods

```
Character(String name, int h, int a, int d, String job)
```
constructer

```
protected void checkStatus()
```
use to check status of characters

```
public void lvUp()
```
level up +1

```
protected abstract void skill()
```
*skill make abstract because 2 types have different skills

```
public void headAdd(Head h)
```
add head armor

```
public void headDrop()
```
drop head armor

```
protected void attack(Character opp)
```
use to attack enemy

```
protected void receiveDmg(int dmg)
```
calculate damage from attacker

Archer Class is sub-class of character

Special variable

```
private int speed = 20;
```

```
public Archer(String name){
    super(name,120,70,30,"Archer");
}
```
Archer's base stat

```
public void checkStatus(){
```
+ add speed stat

```
public void skill()
Skill:  Rain From Paradise
```
:Archer class can charge the attack to make additional damage +20

```
public void attack(Character opp)
```
use to calculate additional dmg, Archer has special ability to change Speed to Atk by Speed/2 + Atk from Skill

Mage Class is sub-class of character

Special variable

```
protected int mana = 100;
```
mana uses for skill and attack

```
public Mage(String name){
    super(name,180,50,40,"Mage");
}
```
Mage's base stat

```
public void checkStatus(){
```
+ add mana stat

```
public void skill()
Skill:  Enlightenment From The Future
```
Mage class can create barrier that Def+20

```
public void attack(Character opp)
```
to calculate additional dmg, Mage must always use Mana-10 to sum with normal atk

```
protected void receiveDmg(int dmg)
```
when Mage's barrier was called from skill. DEF+20 will work

Head Interfaces

```java
public interface Head {
//methods
    void upHead(); //use to level up head
    void checkHeadStatus(); //check status of head
//variables
    String headName();
    int hpAdd();
    int atkAdd();
    int defAdd();
}
```

Mask Class: will give only hp+ and atk+

```java
protected String name;
protected int level;
protected int hpAddition;
protected int atkBonus;
```

Crown Class: will give only hp+ and def+

```java
protected String name;
protected int level;
protected int hpAddition;
protected int defBonus;
```