[Battle Scene:Prototype01: 19.03.17]

[To-do things]
1. Alter interface to hirizontal, switch book <=> battle screen
2. Sample Battle texts

[Problems & Research]
1. enabling/unenabling objects: Book and Text(canvas) => SetActive
2. AI
3.HP bar = slider, make C# class for edit value
4. It seems array comparison doesnt work

[Next to do]
1. Fix the bug (Amber always fails casting)

<class Battle>

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class Battle : MonoBehaviour {
        public int battleID;
        public int enemiesNo;
        public float craftTimeSetting; //can be longer by gathering
runes?

        public bool jasperJoined;
        public bool ioliteJoined;

        public int enemy1HP;
        public int enemy2HP; //input zero if the enemy doesnt
exist!
        public int enemy3HP;
        //i'm too lazy to make enemy class! let's just say this is
not RPG but adventure game =_=...

        private int[] amberInput;
        private float amberSpeed;
        private float craftTime;
        private int turnID;
        private int enemyHPTotal;
        private int playerHPTotal;
        private int jasperPW;
        private int enemyPW;
        private int damage;

        private bool rainning;
        private bool damaging;

        private int turn;
        private int rainEnd;
        private int damEnd;
```

```
        private int damWho;

        private bool next;
        private bool craft;
        private GameObject book;
        private GameObject bui;
        private GameObject bt;

        void Start () {
                next = true; //the obj is tappble or not
                craft = false;
                turnID = 0;
                craftTime = craftTimeSetting;
                jasperPW = 1;
                enemyPW = 1;
                amberSpeed = 1;
                turn = 0;
                rainning = false;
                damaging = false;
                book = GameObject.Find ("Book");
                bui = GameObject.Find ("BattleUI");
                bt = GameObject.Find ("BT");
                book.SetActive (false);
        }

        void Text (string txt) {
                bui.SetActive (true);
                bt.GetComponent<Text>().text = txt;
        }

        void Total () {
                enemyHPTotal = enemy1HP + enemy2HP + enemy3HP;
                playerHPTotal = GameControl.amberHP +
GameControl.jasperHP + GameControl.ioliteHP;
        }

        void Rain() {
                if (enemy3HP > 0) {
                        enemy3HP = enemy3HP - 10;
                }
                if (enemy2HP > 0) {
                        enemy2HP = enemy2HP - 10;
                }
                if (enemy1HP > 0) {
                        enemy1HP = enemy1HP - 10;
                }
                if (GameControl.ioliteHP > 0) {
                        GameControl.ioliteHP =
GameControl.ioliteHP - 10;
                }
                if (GameControl.jasperHP > 0) {
                        GameControl.jasperHP =
GameControl.jasperHP - 10;
                }
```

```
                    if (GameControl.amberHP > 0) {
                            GameControl.amberHP = GameControl.amberHP
- 10;
                    }

                    if (rainEnd == turn) {
                            rainning = false;
                    }
            }

        void Damage () {
                if (damWho == 3 && enemy3HP > 0) {
                        enemy3HP = enemy3HP - 10;
                } else if (damWho == 2 && enemy2HP > 0) {
                        enemy2HP = enemy2HP - 10;
                } else if (damWho == 1 && enemy1HP > 0) {
                        enemy1HP = enemy1HP - 10;
                } else {
                        damaging = false;
                }

                if (damEnd == turn) {
                        damaging = false;
                }
        }

        void Turn () {
                next = false;

                    //Amber's turn
                if (turnID == 0) {
                        bui.SetActive (false);
                        book.SetActive (true);
                        turnID = turnID++;
                        craft = true;

                        //Jasper's turn
                } else if (turnID == 1) {
                        Text ("Jasper attacked an enemy!");
                        if (enemy3HP > 0) {
                                enemy3HP = enemy3HP - 50 *
jasperPW;
                        } else if (enemy2HP > 0) {
                                enemy2HP = enemy2HP - 50 *
jasperPW;
                        } else {
                                enemy1HP = enemy1HP - 50 *
jasperPW;
                        }

                        jasperPW = 1;
                        turnID = turnID++;
                        next = true;
```

```
                } else if (turnID == 2) {
                    if (playerHPTotal <= 210) {
                        Text ("Iolite healed everyone!");
                        //change the rate 0.5 if the game
balance went wrong
                        GameControl.amberHP = (150 -
GameControl.amberHP) / 2 + GameControl.amberHP;
                        GameControl.jasperHP = (150 -
GameControl.jasperHP) / 2 + GameControl.jasperHP;
                        GameControl.ioliteHP = (150 -
GameControl.ioliteHP) / 2 + GameControl.ioliteHP;
                    } else if (GameControl.amberHP <= 30) {
                        GameControl.amberHP = 150;
                    } else if (GameControl.jasperHP <= 30) {
                        GameControl.jasperHP = 150;
                    } else if (GameControl.ioliteHP <= 30) {
                        GameControl.ioliteHP = 150;
                    } else {
                        Text ("Iolite doubled Jasper's
attack.");

                        jasperPW = jasperPW * 2;
                    }

                    turnID = turnID++;
                    next = true;
                } else if (turnID >= 3) {
                    if (GameControl.ioliteHP > 0) {
                        Text ("An enemy attacked
Iolite!");

                        GameControl.ioliteHP =
GameControl.ioliteHP - 30 / enemyPW;
                    } else if (GameControl.jasperHP > 0) {
                        Text ("An enemy attacked
Jasper!");

                        GameControl.jasperHP =
GameControl.jasperHP - 30 / enemyPW;
                    } else {
                        Text ("An enemy attacked Amber!");
                        GameControl.amberHP =
GameControl.amberHP - 30 / enemyPW;
                    }

                    enemyPW = 1;
                    turnID = turnID++;
                    next = true;
                }

        }

        void Update () {
            if (craft == true){
                craftTime = craftTime - 1 *
Time.deltaTime;
                if (craftTime <= 0) {
```

```
                        Text ("Amber failed to cast!");
                        craftTime = craftTimeSetting;
                        book.SetActive (false);
                        next = true;
                } else if (GameControl.casting == true) {
                        amberInput = GameControl.circle;

                        if (amberInput == new int[] { 0,
1, 3 } || amberInput == new int[] { 1, 2, 5 }) {
                                enemyPW = 2;
                                Text ("Amber halfed
enemies' attack powers!");
                        } else if (amberInput == new int[]
{ 0, 1, 5 }) {
                                if (enemy3HP > 0) {
                                        enemy3HP =
enemy3HP - 50;
                                } else if (enemy2HP > 0)
{
                                        enemy2HP =
enemy2HP - 50;
                                } else {
                                        enemy1HP =
enemy1HP - 50;
                                }
                        } else if (amberInput == new int[]
{ 0, 3, 5 }) {
                                if (enemy3HP > 0) {
                                        enemy3HP =
enemy3HP - 50;
                                }
                                if (enemy2HP > 0) {
                                        enemy2HP =
enemy2HP - 50;
                                }
                                if (enemy1HP > 0) {
                                        enemy1HP =
enemy1HP - 50;
                                }
                                if (GameControl.ioliteHP
> 0) {

GameControl.ioliteHP = GameControl.ioliteHP - 50;
                                }
                                if (GameControl.jasperHP
> 0) {

GameControl.jasperHP = GameControl.jasperHP - 50;
                                }
                                if (GameControl.amberHP >
0) {

GameControl.amberHP = GameControl.amberHP - 50;
                                }
```

```
                                            } else if (amberInput == new int[]
{ 0, 3, 4 } || amberInput == new int[] { 0, 4, 5 }) {

                                                    damage = Random.Range
(-100, 0);

                                                    if (enemy3HP > 0) {
                                                            enemy3HP =
enemy3HP + damage;
                                                    }
                                                    if (enemy2HP > 0) {
                                                            enemy2HP =
enemy2HP + damage;
                                                    }
                                                    if (enemy1HP > 0) {
                                                            enemy1HP =
enemy1HP + damage;
                                                    }
                                            } else if (amberInput == new int[]
{ 1, 2, 4 }) {
                                                    amberSpeed = 2;
                                            } else if (amberInput == new int[]
{ 1, 3, 5 }) {
                                                    if (enemy3HP > 0) {
                                                            enemy3HP =
enemy3HP - 50;
                                                    } else if (enemy2HP > 0)
{
                                                            enemy2HP =
enemy2HP - 50;
                                                    } else {
                                                            enemy1HP =
enemy1HP - 50;
                                                    }
                                                    enemyPW = 2;
                                            } else if (amberInput == new int[]
{ 2, 3, 5 }) {

                                                    GameControl.amberHP =
GameControl.amberHP - 25;

                                                    if (enemy3HP > 0) {
                                                            enemy3HP = 0;
                                                    } else if (enemy2HP > 0)
{
                                                            enemy2HP = 0;
                                                    } else {
                                                            enemy1HP = 0;
                                                    }
                                            } else if (amberInput == new int[]
{ 2, 4, 5 }) {
                                                    jasperPW = 0;
                                                    enemyPW = 3;
                                            } else if (amberInput == new int[]
```

```
{ 4, 5, 6 }) {
                                        if (GameControl.amberHP
<= GameControl.jasperHP) {

GameControl.amberHP = 150;
                                        } else if
(GameControl.jasperHP <= GameControl.ioliteHP) {

GameControl.jasperHP = 150;
                                        } else {

GameControl.ioliteHP = 150;

                                        }
                                } else if (amberInput == new int[]
{ 3, 5, 6 }) {
                                        rainning = true;
                                        rainEnd = turn + 3;
                                } else if (amberInput == new int[]
{ 2, 5, 6 }) {
                                        damaging = true;
                                        damEnd = turn + 5;
                                        if (enemy3HP > 0) {
                                                damWho = 3;
                                        } else if (enemy2HP > 0)
{
                                                damWho = 2;
                                        } else {
                                                damWho = 1;
                                        }
                                } else if (amberInput == new int[]
{ 0, 2, 6 }) {
                                        jasperPW = jasperPW * 2;
                                } else {
                                        Text ("Amber failed to
cast!");
                                }

                                GameControl.circle = new int[]
{ 7, 7, 7 };

                                GameControl.casting = false;
                                craftTime = craftTimeSetting *
amberSpeed;

                                book.SetActive (false);
                                next = true;
                        }
                }

                //avoid minus HPs
                if (GameControl.amberHP < 0) {
                        GameControl.amberHP = 0;
                }
                if (GameControl.jasperHP < 0) {
                        GameControl.jasperHP = 0;
                }
```

```
                    if (GameControl.ioliteHP < 0) {
                            GameControl.ioliteHP = 0;
                    }
                    if (enemy3HP < 0) {
                            enemy3HP = 0;
                    }
                    if (enemy2HP < 0) {
                            enemy2HP = 0;
                    }
                    if (enemy1HP < 0) {
                            enemy1HP = 0;
                    }
        }


        void OnMouseDown () {
                if (next == true) {
                    Total ();

                    //skipping dead or non-existing characters
                    if (turnID == 1 && GameControl.jasperHP <=
0) {
                            turnID = turnID++;
                    }

                    if (turnID == 2 && GameControl.ioliteHP <=
0) {
                            turnID = turnID++;
                    }

                    if (turnID == 4 && enemy2HP <= 0) {
                            turnID = turnID++;
                    }

                    if (enemyHPTotal <= 0) {
                            //Winning Text
                            //
Application.LoadLevel("narrativeX");
                    } else if (playerHPTotal == 0 ||
GameControl.amberHP <= 0) {
                            Destroy (this); //only for
prototype!
                            //Gameover text
                            //Application.LoadLevel("start");
                    } else if (enemiesNo < turnID - 2) {
                            turnID = 0;
                            turn = turn++;
                            if (rainning == true) {
                                    Rain ();
                            }
                            if (damaging == true) {
                                    Damage ();
                            }
                            Turn ();
```

```
                                } else {
                                        Turn ();
                                }
                        }
                }
        }

<class HPbar>
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class HPbar : MonoBehaviour {

        public int characterID;
        public int max;
        private int hp;

        void Start () {
                hp = max;
        }

        void Update () {
                this.GetComponent<Slider> ().value = hp;

                if (characterID == 0) {
                        hp = GameControl.amberHP;
                } else if (characterID == 1) {
                        hp = GameControl.jasperHP;
                } else if (characterID == 2) {
                        hp = GameControl.ioliteHP;
                } else if (characterID == 3) {
                        hp =
GameObject.Find("Battle").GetComponent<Battle>().enemy1HP;
                } else if (characterID == 4) {
                        hp =
GameObject.Find("Battle").GetComponent<Battle>().enemy2HP;
                } else if (characterID == 5) {
                        hp =
GameObject.Find("Battle").GetComponent<Battle>().enemy3HP;
                }
        }
}
```