

**LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
MODUL V**



Oleh

Nama : Angelia Regina Dwi Kartika

NIM : 105220039

**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS SAINS DAN KOMPUTER
UNIVERSITAS PERTAMINA
2022**

A. Pendahuluan

Penejelasan mengenai soal Praktikum 5 bahwa Manajer Inora menginginkan sebuah tongkat sihir yang dapat kita atur terlebih dahulu mana nya lalu memiliki fungsi untuk melakukan *summon* terhadap makhluk hidup seperti tumbuhan dan hewan dan ukuran benda yang akan di *summon* harus sesuai. Kemudian tongkat sihir dapat menggerakkan benda mati bukan men-*summon* benda mati dan benda yang telah digerakkan akan masuk ke dalam object benda. Selanjutnya tongkat sihir dapat melakukan *seal ulang* saat pengguna ingin melakukan *summon* kembali terhadap makhluk hidup yang sudah pernah di*summon* sebelumnya. Dengan adanya fungsi *seal ulang* maka spell *summon* tidak akan gagal dan berhasil. Selanjutnya ada spell *undo* yang digunakan untuk kembali ke kondisi tongkat sihir sebelumnya, jika kita tidak ingin men-*summon* dan ingin kembali maka gunakan *spell undo* agar mana dari tongkat sihir tidak berkurang sehingga level mana masih sesuai dengan keadaan sebelumnya. Dalam menggunakan tongkat sihir dibutuhkan segel kontrak dan didalam program akan menjadi sebuah *class* dan berisikan method dari fungsi tongkat yaitu *spell summon*, *spell move object*. Dan tongkat juga memiliki *class* sendiri yang berisikan method dari *spell undo* dan *spell seal ulang*. *Intensitas mana* sangat penting untuk menentukan apakah spell bisa berhasil dilakukan atau tidak dengan tongkat sihir selain itu, pengguna harus menginput sendiri nama makhluk hidup, nama object, ukuran dan juga intensitas mana agar tongkat sihir dapat berjalan dengan baik.

B. Class

No	Nama Class	Deskripsi Singkat
1	Magicwand	Merupakan class yang berisikan beberapa perintah dan berfungsi untuk menghasilkan output yang sesuai terhadap method <i>ContractSeal</i> dan method <i>Undo</i> yang akan dipanggil pada main. Juga menginisiasikan dari <i>arraylist</i> nama benda, ukuran dan mana.
2	App	Merupakan class yang berisikan beberapa perintah dan berfungsi untuk melakukan pemanggilan terhadap value dan juga method <i>ContractSeal</i> , <i>Undo</i> , <i>Re-seal</i> , dan <i>Summon</i> yang telah diinisiasikan dalam <i>class</i> masing-masing. Selain itu, class ini juga menyimpan data yang telah diinput untuk kebutuhan tongkat sihir.
3	Contractseal	Merupakan class yang berisikan beberapa perintah method untuk melakukan <i>summon</i> terhadap makhluk hidup dan juga menggerakkan benda mati. Juga menginisiasikan dari <i>arraylist</i> nama benda, ukuran dan mana.

(Tuliskan Class yang digunakan, tambahkan baris sesuai kebutuhan. Mohon untuk menempatkan class yang digunakan sebagai main class pada baris pertama)

C. Variabel

No	Nama Variabel	Tipe data	Asal Class	Fungsi
1	checkpoint	String	ContractSeal, App	Sebuah nilai untuk melakukan check value terhadap namabenda yang telah diinput untuk menghindari double value.
2	choose	Integer	ContractSeal, App	Sebuah nilai untuk menyimpan inputan dari pilihan yang diberikan disetiap method, agar value input data dapat masuk kedalam kondisi yang sesuai.
3	gohere	Boolean	ContractSeal, App	Tipe data yang memiliki nilai false dan true.
4	NamaBenda	String	ContractSeal, Magicwand, App	Sebuah nilai untuk menyimpan nama dari makhluk hidup dan juga benda mati yang telah diinputkan oleh <i>user</i> dalam <i>arraylist</i> .
5	ManaBenda	String	ContractSeal, Magicwand, App	Sebuah nilai untuk menyimpan besar sebuah mana dalam tongkat sihir untuk melakukan sebuah spell dalam <i>arraylist</i> .
6	UkuranBenda	String	ContractSeal, Magicwand, App	Sebuah nilai untuk menyimpan ukuran atau <i>size</i> dari makhluk hidup dan benda mati untuk menentukan intensitas

				mana yang dikeluarkan dalam <i>arraylist</i> .
7	Manastone	Integer	App	Sebuah nilai untuk menyimpan nilai mana berupa numeral agar memudahkan dalam menentukan hasil akhir dari penggunaan mana.
8	Magical	Integer	App, Magicwand	Sebuah nilai untuk melakukan iterasi dalam perulangan kondisi agar index yang dicari dapat ditemukan.
9	Manaofcreature	Integer	App	Sebuah nilai untuk menentukan mana dari setiap ukuran benda yang disummon atau digerakkan.

(Tuliskan variabel yang digunakan serta asal classnya yang digunakan)

(Jika menggunakan variabel static, mohon tambahkan keterangan static di bagian tipe data, misal static double)

(Tambahkan baris sesuai kebutuhan)

D. Method

No	Nama method	Jenis method	Asal Class	Fungsi
1	SummonLivingCreature	Procedure	ContractSeal	Untuk memberikan inputan data mengenai nama makhluk hidup yang akan disummon, ukuran makhluk hidup dan menentukan berapa banyak mana yang dikeluarkan saat <i>spell summon</i> .

2	Creaturemove	Procedure	ContractSeal	Untuk memberikan inputan data mengenai nama benda yang akan digerakkan, ukuran makhluk hidup dan menentukan berapa banyak mana yang dikeluarkan saat <i>spell move</i> .
3	Boolean Again	Functional	ContractSeal	Untuk melakukan pengecekan lagi terhadap inputan nama benda/makhluk dengan adanya tipe data bool bernilai false
4	Re_seal	Procedure	Magicwand	Untuk melakukan seal ulang terhadap benda/makhluk yang pernah <i>disummon</i> sebelumnya agar bisa <i>disummon</i> lagi.
5	Undo	Procedure	Magicwand	Untuk mengubah nilai terakhir dari mana dan perintah inputan menjadi nilai sebelumnya.
6	MagicSpell	Procedure	App	Untuk menampilkan bahwa spell yang diaktifkan berhasil atau tidak dalam method <i>SummonLivingCreature</i> dan <i>CreatureMove</i> . Method ini bersifat static sehingga seluruh valuenya dapat diakses sebelum object class dibuat.
7	Main	Procedure	App	Untuk melakukan pemanggilan dengan jelas dan tidak ambigu

				terhadap method perintah yang terdapat dalam tongkat sihir.
--	--	--	--	---

(Tuliskan nama method yang digunakan termasuk asal class dimana method itu berada. Jenis method dapat diisi dengan functional atau procedural)

(Jika menggunakan constructor, mohon untuk mencantumkanannya pada n baris awal table ini dengan jenis method : constructor) → n adalah banyaknya class yang digunakan

(Tambahkan baris sesuai kebutuhan)

E. Dokumentasi program

(Mohon untuk menampilkan source code dari setiap class yang digunakan. Cukup dengan menampilkan Class dan method yang ada)

1. ContractSeal

```
import java.util.Scanner;
import java.util.ArrayList;

public class ContractSeal {

    String Checkpoint;
    boolean gohere = false;
    Scanner Magic = new Scanner(System.in);
    int choose;

    ArrayList<String> NamaBenda = new ArrayList<String>();
    ArrayList<String> ManaBenda = new ArrayList<String>();
    ArrayList<String> UkuranBenda = new ArrayList<String>();

    public boolean Again(ArrayList<String> NamaBenda) {
        if (NamaBenda.contains(Checkpoint)) {
            return true;
        }
        gohere = false;
        return false;
    }
}
```

```

public void SummonLivingCreature(ArrayList<String> NamaBenda, ArrayList<String> UkuranBenda, ArrayList<String> ManaBenda) {
    System.out.print(" Input Name of the Creature : ");
    Checkpoint = Magic.nextLine();

    if (Again(NamaBenda)) {
        System.out.println(" Spell Failed, Creature already exist ");
        gohere = true;
    } else {
        gohere = false;
        NamaBenda.add(Checkpoint);
        System.out.println(" Input Size of the Creature : ");
        System.out.println(" 1. Large ");
        System.out.println(" 2. Middle ");
        System.out.println(" 3. Slight ");
        System.out.print(" Please Input Here : ");
        choose = Magic.nextInt();

        if (choose == 1) {
            UkuranBenda.add(" Large ");
            ManaBenda.add(" Mana Large ");
        } else if (choose == 2) {
            UkuranBenda.add(" Middle ");
            ManaBenda.add(" Mana Medium ");
        } else if (choose == 3) {
            UkuranBenda.add(" Slight ");
            ManaBenda.add(" Mana Minimum ");
        }
    }
}

```

```

public void Creatureremove(ArrayList<String> NamaBenda, ArrayList<String> UkuranBenda, ArrayList<String> ManaBenda) {
    System.out.print(" Input Name of the Object : ");
    NamaBenda.add(Magic.nextLine());
    System.out.println(" Input Size of the Object : ");
    System.out.println(" 1. Large ");
    System.out.println(" 2. Middle ");
    System.out.println(" 3. Slight ");
    System.out.print(" Please Input Here : ");
    choose = Magic.nextInt();

    if (choose == 1) {
        UkuranBenda.add(" Large ");
        ManaBenda.add(" Mana Large ");
    }
    else if (choose == 2) {
        UkuranBenda.add(" Middle ");
        ManaBenda.add(" Mana Medium ");
    }
    else if (choose == 3) {
        UkuranBenda.add(" Slight ");
        ManaBenda.add(" Mana Minimum ");
    }
}

```

Didalam *class ContractSeal* merupakan class yang berisikan *arraylist* dari NamaBenda, ManaBenda, dan juga UkuranBenda yang dimana inputan nilai ini akan dimasukkan dan diinisiasikan dalam *arraylist*.

Method Boolean Again untuk melakukan fungsi pengecekan dengan value dari NamaBenda adalah false.

Method SummonLivingCreature sebuah fungsi untuk melakukan spell *summon* dengan menentukan nama benda, size benda dan juga mana yang digunakan.

Method Creaturemove sebuah fungsi untuk melakukan spell *move* menggerakkan benda mati dengan menentukan inputan dari nama benda, size benda, dan juga mana yang digunakan.

2. *Magicwand*

```
import java.util.ArrayList;
import java.util.Scanner;

public class Magicwand {

    ArrayList<String> NamaBenda = new ArrayList<String>();
    ArrayList<String> ManaBenda = new ArrayList<String>();
    ArrayList<String> UkuranBenda = new ArrayList<String>();
    Scanner Magic = new Scanner(System.in);

    ContractSeal passMagical = new ContractSeal();

    public void Re_seal ( ArrayList<String> NamaBenda, ArrayList<String> UkuranBenda, ArrayList<String> ManaBenda, int Magical ) {
        NamaBenda.remove(Magical - 1);
        UkuranBenda.remove(Magical - 1);
        ManaBenda.remove(Magical - 1);
        Magical--;
        System.out.println(" All Success!! ");
    }

    public void Undo ( ArrayList<String> NamaBenda, ArrayList<String> UkuranBenda, ArrayList<String> ManaBenda, int Magical ) {
        NamaBenda.remove(Magical - 1);
        UkuranBenda.remove(Magical - 1);
        ManaBenda.remove(Magical - 1);
        System.out.println(" Undo Spell Success!! ");
    }

}
```

Didalam *class Magicwand* merupakan class yang berisikan *arraylist* dari NamaBenda, ManaBenda, dan juga UkuranBenda yang dimana inputan nilai ini akan dimasukkan dan diinisiasikan dalam *arraylist*.

Method *Re_seal* untuk melakukan fungsi yang dimana dilakukan iterasi untuk mencari index yang telah menyimpan value dari inputan data yang ingin kita summon kembali, agar tidak bertabrakan dan membuat nilai value yang sama dalam satu program.

Method *Undo* menggunakan *remove* untuk menghapus data sebelumnya sehingga nilai inputan terakhir akan terhapus jadi nilai mana dan juga nama benda kembali ke nilai sebelumnya.

3. App

```
import java.util.ArrayList;
import java.util.Scanner;
public class App {

    static int Manastone;
    static int Magical = 0;
    static ArrayList<String> NamaBenda = new ArrayList<String>();
    static ArrayList<String> ManaBenda = new ArrayList<String>();
    static ArrayList<String> UkuranBenda = new ArrayList<String>();

    Scanner Magic = new Scanner(System.in);

    public static void MagicSpell(int input1, int input2) {
        if (input1 == input2 || input1 > input2) {
            System.out.println(" ~~~~~ Abracadabra Cling Cling ~~~~~ ");
        }

        else if (input1 < input2) {
            System.out.println(" Spell is Wrong ");
        }
    }
}
```

Run | Debug

```
public static void main(String[] args) throws Exception {
    Scanner Magic = new Scanner(System.in);
    int ManaofMana;
    int choose = 0;

    while (choose != 8) {
        System.out.println(" _____ ");
        System.out.println(" ===== One stick to summon them all ===== ");
        System.out.println(" ~~~~~");
        System.out.println(" Dashboard : ");
        System.out.println(" 1. Set Mana of Wand ");
        System.out.println(" 2. Summon Creature ");
        System.out.println(" 3. Move Object ");
        System.out.println(" 4. Undo ");
        System.out.println(" 5. Seal Again ");
        System.out.println(" 6. Last Condition of Wand ");
        System.out.println(" 7. Log out ");
        System.out.println(" _____ ");
        System.out.println(" \n ");
        System.out.print(" Please Input Here : ");
        choose = Magic.nextInt();
        Magic.nextLine();

        ContractSeal Castspell = new ContractSeal();
        Magicwand Wand = new Magicwand();

        if (choose == 1) {
            System.out.println(" ===== Set Mana ===== ");
            System.out.println(" 1. Maximun ");
            System.out.println(" 2. Medium ");
            System.out.println(" 3. Minimum ");
            System.out.print(" Please Input Here : ");

            ManaofMana = Magic.nextInt();
            Magic.nextLine();

            if (ManaofMana == 1) {
                Manastone = 100;
            } else if (ManaofMana == 2) {
                Manastone = 50;
            } else if (ManaofMana == 3) {
                Manastone = 20;
            }
        } else if (choose == 2) {
            String Checkpoint;

            int Manaofcreature = 0;
```

```

if (NamaBenda.isEmpty()) {
    Castspell.SummonLivingCreature(NamaBenda, UkuranBenda, ManaBenda);
    if (UkuranBenda.get(Magical) == " Large ") {
        Manaofcreature = 100;
    } else if (UkuranBenda.get(Magical) == " Middle ") {
        Manaofcreature = 50;
    } else if (UkuranBenda.get(Magical) == " Slight ") {
        Manaofcreature = 20;
    }
}

```

```

    MagicSpell(Manastone, Manaofcreature);
    System.out.println(" ===== Summon Creature =====");
    System.out.println(" Input Name of the Creature : ");
    System.out.println(" Creature Name : " + NamaBenda.get(Magical));
    System.out.println(" Level of size : " + UkuranBenda.get(Magical));

    Magical++;
} else {
    System.out.println(NamaBenda.size());
    Castspell.SummonLivingCreature(NamaBenda, UkuranBenda, ManaBenda);
    System.out.println(Castspell.Checkpoint);
    if (Castspell.Again(NamaBenda) && Castspell.gohere) {
        System.out.println(" Welcome Back ");
    } else {
        if (UkuranBenda.get(Magical) == " Large ") {
            Manaofcreature = 100;
        } else if (UkuranBenda.get(Magical) == " Middle ") {
            Manaofcreature = 50;
        } else if (UkuranBenda.get(Magical) == " Slight ") {
            Manaofcreature = 20;
        }

        MagicSpell(Manastone, Manaofcreature);
        System.out.println(" ===== Summon Creature =====");
        System.out.println(" Input Name of the Creature : ");
        System.out.println(" Creature Name : " + NamaBenda.get(Magical));
        System.out.println(" Level of size : " + UkuranBenda.get(Magical));

        Magical++;
    }
}
}

```

```

else if (choose == 3) {
    Castspell.Creaturemove>NamaBenda, UkuranBenda, ManaBenda);
    int Manaofcreature = 0;
    if (UkuranBenda.get(Magical) == " Large ") {
        Manaofcreature = 100;
    } else if (UkuranBenda.get(Magical) == " Middle ") {
        Manaofcreature = 50;
    } else if (UkuranBenda.get(Magical) == " Slight ") {
        Manaofcreature = 20;
    }

    MagicSpell(Manastone, Manaofcreature);
    System.out.println(" ===== Move Object =====");
    System.out.println(" Input Name of the Object : ");
    System.out.println(" Objectc Name : " +>NamaBenda.get(Magical));
    System.out.println(" Level of size : " +>UkuranBenda.get(Magical));

    Magical++;
}

```

```

else if (choose == 4) {
    System.out.println(" ===== Undo =====");
    if (ManaBenda.get (Magical - 1) == " Maximum ") {
        Manastone = 100;
        System.out.println (" Mana has Been Upgrade into Maximum ");
    }

    else if (ManaBenda.get(Magical - 1) == " Medium ") {
        Manastone = 50;
        System.out.println (" Mana has Been Upgrade into Medium ");
    }

    else if (ManaBenda.get(Magical - 1) == " Minimum ") {
        Manastone = 20;
        System.out.println (" Mana has Been Upgrade into Minimum ");
    }
    Wand.Undo>NamaBenda, UkuranBenda, ManaBenda, Magical);

    Magical--;
}

else if (choose == 5) {
    Wand.Re_seal>NamaBenda, UkuranBenda, ManaBenda, Magical);

    Magical--;
}

```

```

else if (choose == 6) {
    System.out.println(" ===== Lat Condition of Wand =====");
    System.out.println (" Level of Mana = " + Manastone);
}

else {
    System.out.println(" =====");
    System.out.println(" ===== Hope It Will Work =====");
    System.out.println(" ===== because coding is fun ^_^ =====");
    System.out.println(" =====");
    break;
}
}
}
}

```

Didalam *class App* berisikan perintah pemanggil dari 7 menu yang ada terhadap method di classnya masing-masing agar menghasilkan output yang diinginkan dan sesuai permintaan manager inora.

F. Kesimpulan

Untuk menyelesaikan permintaan dari manager inora digunakan *arraylist* untuk variabel nama benda, ukuran benda, dan juga mana. Dalam melakukan *spell summon* dan juga *spell move* digunakan iterasi yang berfungsi untuk mencari index yang menyimpan vaue dari inputan nama agar dapat dipanggil kembali, digerakkan atau di lakukan *spell seal ulang*. Tanpa adanya iterasi berulang maka program yang telah dibuat tidak dapat berjalan dan mengidentifikasi value yang ingin dicari (error). Dengan adanya penggunaan *arraylist* dalam program ini membantu program menyimpan dan mencari banyak data dengan mudah.