Planning:

Pseudo code / Algorithm

```
public class AdventureModel {
 private Room rooms[];
 private Thing things[];
 private Adventurer adventurer;
 private Wall walls[];
 private Water waters[];
```

Items

```
Item[0] = new Items("Sword", "A short sword");
 Item[1] = new Items("Bow", "An Oak longbow");
Item[2] = new Items("Arrow", "Basically useless without a bow");
 Item[3] = new Items("Coin",   "Gold? maybe a merchant needs that");
 Item[4] = new Items("Key", "Hmm whats the key for?");
 Item[5] = new Items("DemonBone ", "The remains of a demon");
Item[6] = new Items("Lamp", "A light source");
Item[7] = new Items("Potion", "Must need for an adventure");
Item[8] = new Items("Shield", "Block Damage");
Item[9] = new Items("Necklace", "Necklace of Hope");
 Item[10] = new Items("Key", "A rusty key");
```

Menu() {

```
 System.out.println("Choose command: ");
 System.out.println("Go");
 System.out.println("Look");
 System.out.println("Take");
 System.out.println("Drop");
 System.out.println("Use");
 System.out.println("Exit ");
 System.out.println();
    }
```

Interaction with items

Can get items

Use array with items ArrayList<Items>();

Drop items inventory.remove(item);

Add rooms

Use array with items ArrayList<Items>();

```
rooms[1] = new Room("Town Square", "In the middle of your home town. ");
 rooms[2] = new Room("A Road", "Leads to the unknown. ");
 rooms[3] = new Room("SplitRoad","Three road either leading to Forest, DarkCrypt and Graveyard. ");
 rooms[4] = new Room("Forest","A narrow track wound steeply up through dense forest.
 rooms[6] = new Room("Tavern","A resting area for the hero and start postion. ");
 rooms[11] = new Room("King's Landing(End)","King's Landing(End). ");
 rooms[5] = new Room("AbandonedHouse","A dark house. ");
 rooms[7] = new Room("Graveyard","For the dead. ");
 rooms[7].addObject(Item[10])
 rooms[8] = new Room("Chest Room","TREASURE, well not really. ");
 rooms[8].addObject(Item[0])
 rooms[9] = new Room("DarkCrypt","Really creepy, need a Lamp to brighten the area. ");
 rooms[9].addObject(Item[5]);
 rooms[10] = new Room("Dungeon","The darkest dungeon. ");
 rooms[10].addObject(Item[3]);
```

Also add items inside of rooms

Start new adventure

```
 AdventureModel adv = new AdventureModel();
 adv.startGame();
```

Walls/Water

Rooms if walls[0].addDirection("East", rooms[1]); same with waters using an array

```
 for(int i =0 ;i< 4 ;i++)
 walls[i] = new Wall();

 for(int i =0 ;i< 10 ;i++)
 waters[i] = new Water();
```