1. Package base
   1. Class Banknote

1.1.1 Fields

|  |  |
| --- | --- |
| **- int** banknoteValue | มูลค่าของธนบัตร |
| - **int** amount | จำนวนธนบัตร |

1.1.2 Constructor

|  |  |
| --- | --- |
| + **public** Banknote(**int** banknoteValue) | -รับมูลค่าของธนบัตร  -กำหนดจำนวนธนบัตรเริ่มต้นที่ 1 ใบ โดยใช้ setter |

1.1.3 Methods

|  |  |
| --- | --- |
| + **int** getAmount() | คืนจำนวนธนบัตร |
| + **void** setAmount(**int** amount) | กำหนดค่าธนบัตร ให้มีค่ากับ amount  โดยถ้ารับจำนวนธนบัตรเข้ามามีค่าน้อยกว่า 0 จะกำหนดจำนวนธนบัตรเป็น 0 |
| + **int** getBanknoteValue() | คืนมูลค่าของธนบัตร |
| + **void** setBanknoteValue(**int** banknoteValue) | กำหนดมูลค่าของธนบัตรให้มีค่าเป็น banknoteValue ที่รับเข้ามา |
| + **boolean** equals(Banknote other) | เปรียบเทียบ**มูลค่าธนบัตร**ว่าเท่ากันหรือไม่ |

* 1. Class CasinoBudget

1.2.1 Fields

|  |  |
| --- | --- |
| - ArrayList<Banknote> banknoteTypeList | -ArrayList เก็บ Banknote |

1.2.2 Constructor

|  |  |
| --- | --- |
| + CasinoBudget() | - สร้าง ArrayList เก็บ Banknote มูลค่า 10,000 20,000 30,000 40,000 50,000 60,000 70,000 และ 80,000 อย่างละ 1 ใบ |

1.2.3 Methods

|  |  |
| --- | --- |
| + ArrayList<Banknote> getBanknoteTypeList() | -คืน ArrayList ที่เก็บ Banknote โดยก่อนคืนArrayList จะทำการสลับลำดับของ Banknote  แบบสุ่ม |

* 1. Dice

1.3.1 Fields

|  |  |
| --- | --- |
| - **String** colour | -สีของลูกเต๋า |
| - **int** point | -แต้มของลูกเต๋าที่หงายหน้าอยู่ในขณะนั้น |

1.3.2 Constructor

|  |  |
| --- | --- |
| + Dice(String colour) | -รับสีของลูกเต๋าเข้ามา -กำหนดสีของลูกเต๋าเหมือนกับ colour ที่รับเข้ามา   โดยใช้ **setter** -กำหนดหน้าลูกเต๋าที่หงายขณะนั้นให้มีค่าเป็น 1  โดยใช้ setter |

1.3.3 Method

|  |  |
| --- | --- |
| + **void** rolling() | - สุ่มหน้าลูกเต๋าระหว่างเลข 1-6 และ กำหนดหน้า  ลูกเต๋าที่หงายอยู่ขณะนั้นให้เป็นเลขที่สุ่มได้ |
| + **String** getColour() | - คืนสีของลูกเต๋า |
| + **void** setColour(String colour) | - กำหนดสีของลูกเต๋าให้เหมือน colour ที่รับเข้ามา |
| + **int** getPoint() | - คืนหน้าของลูกเต๋าที่หงายอยู่ในขณะนั้น |
| + **void** setPoint(**int** point) | - กำหนดหน้าของลูกเต๋าที่หงายอยู่ในขณะนั้นให้มีค่าเท่ากับ point ที่รับเข้ามา |

* 1. Class Location

1.4.1 Fields

|  |  |
| --- | --- |
| - ArrayList<**Banknote**> fund | - ArrayList เก็บธนบัตรทั้งหมดของ Location |
| - ArrayList<**Integer**> diceInLocation | - ArrayList เก็บจำนวนลูกเต๋าที่ผู้เล่นวางไว้ใน  Location โดยเริ่มที่ ArrayList ช่องที่ 0 แทน  จำนวนลูกเต๋าที่วางไว้บน Location ของ ผู้เล่น  คนที่ 1 จนถึงช่องสุดท้ายแทน จำนวนลูกเต๋าที่วาง  ไว้บน Location ของผู้เล่นคนสุดท้าย |
| - **String** name | - ชื่อ Location |
| - **CasinoBudget** budget | - เงินกองกลางของเกม |
| - **int** amountOfPlayer | - จำนวนผู้เล่นทั้งหมด |
| - **int** diceValue | - ค่าของลูกเต๋าประจำ Location |

1.4.2 Constructor

|  |  |
| --- | --- |
| -Location(String name,**int** amountOfPlayer, **int** diceValue) | -รับ ชื่อสถานที่,จำนวนผู้เล่น และ หน้าลูกเต๋าประจำ  Location ตามลำดับ  - กำหนดชื่อสถานที่ให้เหมือน name ที่รับเข้ามา  โดยใช้ setter  -กำหนดจำนวนผู้เล่นให้เหมือน amountOfPlayer  ที่รับเข้ามาโดยใช้ setter  -กำหนดหน้าลูกเต๋าประจำ Location ให้เหมือน  diceValue ที่รับเข้ามาโดยใช้ setter  -สร้าง ArrayList<Banknote> เพื่อใช้เก็บธนบัตร  ทั้งหมดที่ Location มี  -สร้าง ArrayList<Integer> เพื่อเก็บลูกเต๋าของผู้เล่น  แต่ละคนที่วางไว้บน Location โดยมีช่องเท่ากับ  จำนวนผู้เล่นและทุกช่องมีค่าเป็น 0  - ตรวจสอบจำนวนเงินใน Location ว่ามีค่าน้อยกว่า  50,000 หรือไม่ โดย สามารถหาผลรวมของจำนวน  เงินที่ Location มีได้ โดยเรียก  **fundValue(fund)** ถ้าน้อยกว่าให้เพิ่มธนบัตรให้  Location เพิ่มโดยการเรียกใช้ **updateFund()**  จนกว่าจะมี จำนวนเงิน มากกว่าหรือเท่ากับ 50,000 |

1.4.3 Method

|  |  |
| --- | --- |
| + **int** fundValue(ArrayList<Banknote> fund) | - คืนผลรวมของมูลค่าธนบัตรทั้งหมดที่ Location มี |
| + **void** addDice(Player p,int amount) | -รับ Player ที่ต้องการวางลูกเต๋า และ   จำนวนลูกเต๋าที่ต้องการจะวาง  - เพิ่มจำนวนลูกเต๋าที่ผู้เล่นวางบน Location โดย  ตรวจสอบสีลูกเต๋าของผู้เล่น   ถ้าลูกเต๋า**สีขาว** แสดงถึงผู้เล่นคนแรก ให้ใส่ลูกเต๋า  ลงในช่องที่ 0 ของ diceInLocation  ถ้าลูกเต๋า**สีน้ำเงิน** แสดงถึงผู้เล่นคนที่ 2 ให้ใส่  ลูกเต๋า ลงในช่องที่ 1 ของ diceInLocation  ถ้าลูกเต๋า**สีแดง** แสดงถึงผู้เล่นคนที่ 3 ให้ใส่ลูกเต๋า  ลงในช่องที่ 2 ของ diceInLocation  ถ้าลูกเต๋า**สีเขียว** แสดงถึงผู้เล่นคนที่ 4 ให้ใส่ลูกเต๋า  ลงในช่องที่ 3 ของ diceInLocation  ถ้าลูกเต๋าสีเหลือง แสดงถึงผู้เล่นคนที่ 5 ให้ใส่  ลูกเต๋า ลงในช่องที่ 4 ของ diceInLocation |
| + **boolean** haveSameElement(**int** amount) | - รับจำนวนของลูกเต๋าเข้ามา  - ตรวจสอบว่า ใน diceInLocation มีค่าซ้ำกัน  หรือไม่ ถ้าซ้ำให้เปลี่ยนค่าทุกช่องที่มีค่าเท่ากับ  amount เป็น 0 และคืน false ถ้าไม่ซ้ำไม่ต้องทำ  อะไรและคืน true |
| + **int** sendReward(Player p) | - ให้เงินจำนวนเท่ากับมูลค่าของธนบัตรที่มีมูลค่า  มากที่สุดแก่ Player p และนำธนบัตรใบนั้นออก  ทำการตรวจสอบจำนวนธนบัตรถ้าธนบัตรมูลค่า  ใดมีจำนวนเป็น 0 ให้ทำการลบออกจากArrayList  และทำการเรียง ธนบัตรจากมูลค่ามากไปน้อย |
| + **void** updateFund() | - จะถูกเรียกในกรณีที่ผลรวมมูลค่าธนบตรทั้งหมดมี  ค่าน้อยกว่า 50,000 จะทำการเพิ่มธนบัตรใบแรก  สุดให้ใน budget ให้กับ Location ก่อนจะ  ทำการเพิ่มจะเช็คว่ามีธนบัตรมูลค่าเท่ากับ  ใบแรกสุดใน budget ใน fund หรือไม่   ถ้ามีให้ทำการเพิ่มจำนวนของธนบัตรมูลค่านั้น  ถ้าไม่มีให้ทำการสร้าง Banknote ที่มีมูลค่าเท่ากับ  ธนบัตรใบแรกสุดให้ใน budget  แล้วเพิ่มเข้าไปในfund - ทำการเรียงมูลค่าธนบัตรใน fund จากมูลค่า  มากไปน้อย |
| + **int** getAmountOfPlayer() | - คืนค่าจำนวนผู้เล่น |
| + **void** setAmountOfPlayer(**int** amountOfPlayer) | - กำหนดจำนวนผู้เล่นให้มีค่าเหมือน  amountOfPlayer  ถ้าamountOfPlayer น้อยกว่า 2 ให้กำหนดเป็น 2  ถ้ามากกว่า 5 ให้กำหนดเป็น 5 |
| + ArrayList<Banknote> getFund() | - คืน ArrayList ที่เก็บธนบัตรทั้งหมดใน Location |
| + **void** setFund(ArrayList<Banknote> fund) | - กำหนด ArrayListที่เก็บธนบัตรทั้งหมดให้  Location ให้เหมือน fund ที่รับเข้ามา |
| + ArrayList<Integer> getDiceInLocation() | - คืน ArrayList ที่เก็บลูกเต๋าของแต่ละคนที่วาง  ไว้บน Location |
| + **void** setDiceInLocation(ArrayList<Integer> diceInLocation) | - กำหนด ArrayListที่เก็บธนบัตรทั้งหมดให้  Location ให้เหมือน diceInLocation  ที่รับเข้ามา |
| + String getName() | - คืนชื่อของ Location |
| + **void** setName(String name) | - ตั้งชื่อของ Location ให้เหมือนกับ name ที่  รับเข้ามา ถ้ารับ Stringว่างเข้ามาให้ตั้งชื่อสถานที่  เป็น “Location” |
| + **int** getDiceValue() | - คืนค่าของลูกเต๋าประจำ Location |
| + **void** setDiceValue(**int** diceValue) | - กำหนดหน้าลูกเต๋าประจำ Location ให้มีค่า  เหมือน diceValue ที่รับเข้ามา |

* 1. Class SpecialLocation extends Location

1.5.1 Constructor

|  |  |
| --- | --- |
| +SpecialLocation(String name ,  **int** amountOfPlayer,**int** dicevalue) | -รับ ชื่อสถานที่,จำนวนผู้เล่น และ หน้าลูกเต๋าประจำ  Location ตามลำดับ  - กำหนดชื่อสถานที่ให้เหมือน name ที่รับเข้ามา  โดยใช้ setter  -กำหนดจำนวนผู้เล่นให้เหมือน amountOfPlayer  ที่รับเข้ามาโดยใช้ setter  -กำหนดหน้าลูกเต๋าประจำ Location ให้เหมือน  diceValue ที่รับเข้ามาโดยใช้ setter  -สร้าง ArrayList<Banknote> เพื่อใช้เก็บธนบัตร  ทั้งหมดที่ Location มี  -สร้าง ArrayList<Integer> เพื่อเก็บลูกเต๋าของผู้เล่น  แต่ละคนที่วางไว้บน Location โดยมีช่องเท่ากับ  จำนวนผู้เล่นและทุกช่องมีค่าเป็น 0  - ตรวจสอบจำนวนเงินใน Location ว่ามีค่าน้อยกว่า  50,000 หรือไม่ โดย สามารถหาผลรวมของจำนวน  เงินที่ Location มีได้ โดยเรียก  **fundValue(fund)** ถ้าน้อยกว่าให้เพิ่มธนบัตรให้  Location เพิ่มโดยการเรียกใช้ **updateFund()**  จนกว่าจะมี จำนวนเงิน มากกว่าหรือเท่ากับ 50,000 |

1.5.2 Method

|  |  |
| --- | --- |
| +**int** sendReward(Player p) | - ให้เงินจำนวนเท่ากับ2เท่าของมูลค่าของธนบัตรที่มี  มูลค่า มากที่สุดแก่ Player p และนำธนบัตร  ใบนั้นออก  ทำการตรวจสอบจำนวนธนบัตรถ้าธนบัตรมูลค่า  ใดมีจำนวนเป็น 0 ให้ทำการลบออกจากArrayList  และทำการเรียง ธนบัตรจากมูลค่ามากไปน้อย |