1. Продумать классы, предметную область, свойства, данные, их взаимодействие
2. Еще не успел доделать немного классы зданий, но основное есть.

\*Можно подумать над использованием графа для выбора бандитами наилучших зданий в городе. Например, бандиты могут ограбить только 3 соседних (или не соседних) здания и надо выбрать лучшие

**Классы:**

* GameSetting – класс всей игры. Содержит двумерный массив Building[,] map всех клеток игры. Каждая клетка является наследником Entity. Если на клетке ничего нет, то null.

Есть поле int sheriffs – значение варьируется от 0 до 2. Если значение == 2, то нельзя строить SheriffsHouse.

Имеет поля AttackTimer и NewCitizienTimer, от которых зависит время атаки / время появление нового жителя.

Имеет поле Citiziens citiziens.

* Entity – общий класс для всех «живых», взаимодействующих с миром сущностей. Имеет конструктор Entity, в который принимается координата клетки на map. Содержит публичное поле Location, имеет метод Delete для удаления с карты (превращает клетку Location в null).
* Citiziens – класс жителей. Имеет поле HashSet<int> citiziens, хранящий id всех жителей. Имеет поле int citiziensLimit, определяющий максимальный размер citiziens. По умолчанию citiziensLimit = 4, citiziens содержит 1 жителя с id = 0.

Имеет методы Add(int id), добавляющий жителя; Remove(int id), удаляющий жителя; Increase(int number), увеличивающий citiziensLimit.

* Bandits – наследует Entity. Имеет поле Building[] buildings. В конструкторе задается размер buildings = 3 – sheriffs (sheriffs хранится в GameSetting). Имеет метод FindBuildings – ищет оптимальные здания для грабежа, FindWay - выстраивает оптимальный маршрут по зданиям из buildings.
* Building – наследует Entity. Имеет поля int Happiness (обозначает время появления нового жителя), int Worker, int BudgetWeakness (измеряется в %), (int,int) Location. Если Worker < 0, то работника в здании нет. Happiness может быть любого знака. BudgetWeakness >= 0 всегда. В конструкторе задаются все эти параметры (Location = индексу в массиве map). Имеет виртуальный метод Delete(), который по умолчанию заменяет в map ссылку на null и вычитает из поля NewCitizienTimer GameSetting текущий Happiness.

Будут классы LivingHouse (переопределяет delete), SheriffsHouse (переопределяет delete), Shop, Saloon, Jail, RailroadStation, который наследуют Building.