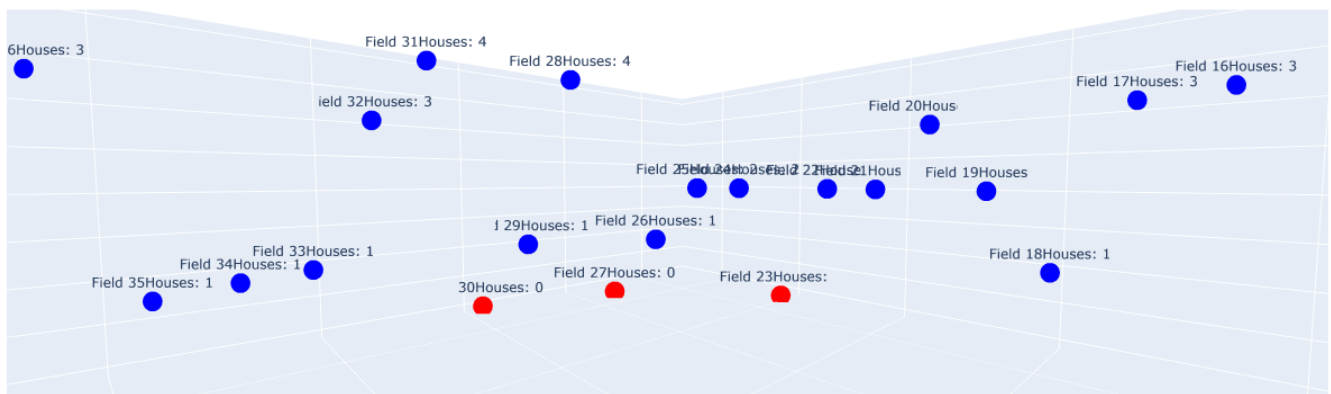
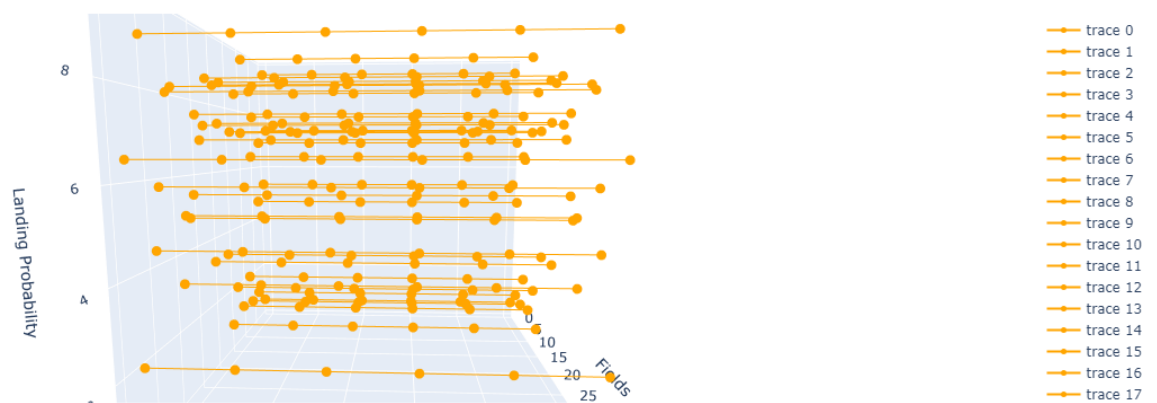


## Visualization

3D Model of Monopoly Board



3D Landing Probability on Monopoly Fields



## Description:

### Индикаторная Функция

```
def player_state_indicator(position, balance):
    return {
        'on_field': True if position >= 0 else False,
        'in_jail': position == 10,
        'on_go': position == 0,
        'on_free_parking': position == 20,
        'on_community_chest': position in [2, 17],
```

```

        'on_chance': position in [7, 22, 36],
        'balance': balance
    }

```

#### **Функция *player\_state\_indicator*:**

- принимает текущую позицию игрока и его баланс и возвращает словарь, описывающий состояние игрока:
  - *on\_field*: указывает, находится ли игрок на поле.
  - *in\_jail*: указывает, находится ли игрок в тюрьме (позиция 10).
  - *on\_go*: указывает, находится ли игрок на поле "Go".
  - *on\_free\_parking*: указывает, находится ли игрок на поле «Бесплатная парковка».
  - *on\_community\_chest*: указывает, находится ли игрок на одном из полей «Сундука сообщества».
  - *on\_chance*: указывает, находится ли игрок на одном из полей «Шанс».
  - *balance*: возвращает текущий баланс игрока.

#### **Функция *simulate\_game*:**

- Эта функция моделирует игру «Монополия» на заданное количество ходов (*num\_turns*).
- Создается объект *board*, чтобы получить доступ к полям игры.
- *position* инициализируется значением 0 (начальная позиция игрока).
- *balance* инициализируется значением 10 (начальный баланс игрока).
- *win\_count* используется для подсчета количества выигрышей игрока.

#### **Цикл Симуляции:**

- Цикл выполняется *num\_turns* раз, имитируя каждый ход игрока.
- В каждой итерации выполняется бросок двух кубиков с использованием *np.random.randint(1, 7)* для получения случайных чисел от 1 до 6 и их суммирования.

#### **Объяснение кода 3D-модели поля**

1. **Координаты полей:** используем полярные координаты, чтобы разместить поля по кругу. *radius* определяет радиус круга.
2. **Состояние зданий:** используем случайные значения от 0 до 4 для представления количества домов на каждом поле.
3. **Создание графика:** используем *Scatter3d* для отображения полей в виде точек, добавляя текст с номером поля и количеством домов.
4. **Настройка графика:** Настраиваем заголовок и оси для лучшего восприятия.

---

#### **ОБЪЯСНЕНИЕ КОДА 3D-СЕТКИ ВЕРОЯТНОСТЕЙ**

1. **Данные вероятностей:** создаем случайные вероятности для каждого поля, умножая их на 10 для улучшения визуализации.

2. **Создание графика:** используем Scatter3d для добавления данных.  $x$  представляет номера полей,  $y$  — значения бросков кубиков, а  $z$  — вероятность попадания на соответствующее поле.
3. **Настройка графика:** настраиваем заголовок и оси, чтобы график был понятен.

---

#### ОБЪЯСНЕНИЕ КОДА 3D-МОДЕЛИ ПОЛЯ

##### 1. Координаты полей:

- **Дискретные координаты:** Мы используем полярные координаты для расположения 40 полей по кругу. Это демонстрирует работу с дискретными элементами, где каждое поле занимает фиксированное место, создавая конечное и четко определенное множество.

##### 2. Состояние зданий:

- **Случайные величины:** Количество домов на каждом поле задается случайными целыми числами от 0 до 4. Этот подход иллюстрирует применение теории вероятностей — важного раздела дискретной математики, позволяющего анализировать влияние случайных величин на состояние полей.

##### 3. Создание графика:

- **Дискретные точки:** Используя Scatter3d, мы представляем каждое поле как отдельную точку в 3D-пространстве. Это подчеркивает дискретный характер полей, где каждое поле является уникальным элементом в наборе.

##### 4. Настройка графика:

- **Упорядоченные наборы данных:** Набор координат четко определен и упорядочен, что является основой работы с дискретными структурами. Это позволяет визуализировать данные так, чтобы выявить закономерности.

---

#### ОБЪЯСНЕНИЕ КОДА 3D-СЕТКИ ВЕРОЯТНОСТЕЙ

##### Данные вероятностей:

- **Вероятностное распределение:** Случайные вероятности для каждого поля демонстрируют использование вероятностных распределений, что является важным аспектом дискретной математики.

##### Создание графика:

- **Дискретные значения:** Номера полей представлены как дискретные значения (от 1 до 40), а вероятности и значения бросков кубиков являются дискретными величинами. Это позволяет визуализировать взаимодействия между различными элементами игры.

##### Настройка графика:

- **Определение осей:** Настройка графиков с использованием дискретных значений для осей (например, количество бросков и поля) дает возможность проводить анализ и делать выводы о вероятностях приземления.