OVERVIEW OF THE PROBLEM:

The aim of this project is to develop a generic markdown pricing strategy for a retailer.

Initial Stock: 2000 Units
- Demand is difficult to predict and there is no restocking.

Initial Price: $60
- **Possible markdowns**: 10% ($54), 20% ($48), 40% ($36)
- When the price of an item is marked down, it cannot be raised; only maintained or decreased

Timeframe: 15 week selling season
- Leftover inventory after this timeline has no salvage value.
-

Goal: Maximize revenues from the 2000 units.

APPROACH

To create a strategy, we chose to view this task as an optimization problem, and created a **Non-Linear Programming Model** using Gurobi.

It is important to note that the primary goal is to maximize revenue, not minimize leftover inventory. As such, it is inevitable that there may be some scenarios where not all of the inventory will be sold out in order to achieve the optimal revenue.

DEMAND CALCULATIONS

Certain key considerations regarding the demand are:

- Demand for the product is uncertain. This means that:
    - Demand could vary across different weeks even at the same price point.
    - Demand curves for different items vary

Modeling the demand uncertainty is difficult because:

- We have limited data (no product information, selling season is quite short so we cannot obtain a robust trend)

Because of these uncertainties, we made the following **assumptions** when building our model:

1. Demand is constant at a given price.
2. The absolute demand varies greatly from item to item but the relative lift from a price drop is consistent across items.

Using historical data from 15 items, we calculated the average weekly sales quantities at each price (Table 1 - blue part). Using these average sales values, we calculated the relative demand jump from the initial price to each markdown price (Table 1 - green part). Finally, we obtained the average demand jump from the initial price of $60 to each markdown price (Table 1 – red part). This average

jump value serves as a multiplier when calculating the average demand at each price point, which is implemented into our optimization model.

Average Weekly Sales at Different Prices				Relative Demand Jump	Average Jump
60.00	54.00	48.00	36.00		
58.33	76.00			1.30	
107.67	144.00			1.34	
59.33	82.33			1.39	1.31
61.17	77.89			1.27	
92.50	113.67			1.23	
114.14		150.13		1.32	
67.43		119.63		1.77	
53.00		96.75		1.83	1.73
73.71		131.88		1.79	
67.29		97.13		1.44	
100.44			178.71	1.78	
64.11			166.57	2.60	
65.56			177.29	2.70	2.81
61.44			147.14	2.39	
62.33			156.29	2.51	

Table 1: Calculation of Relative Demand Jump due to the Implementation of each Markdown Price

OPTIMIZATION MODEL

To build our NLP model, we defined the following decision variables, objective function and constraints:

Decision Variables

$$\boldsymbol{P_i \text{ where } i = \{1, 2, 3 \dots 15\}} \textit{ (price charged on a given week)}$$

$$\boldsymbol{D_i \text{ where } i = \{1, 2, 3 \dots 15\}} \textit{ (sales quantity on a given week)}$$

$$\boldsymbol{X_{ij} \text{ where } i = \{1, 2, 3 \dots, 15\} \text{ and } j = \{1, 2, 3, \dots, 15\}} \textit{ (Dummy variable relating to price and sale)}$$

Objective Function

$$Maximixe \sum_{i=1}^{15} Di * Pi$$

Constraints

1. Initial sale quantity and price is known

$$D_1 = x \quad (x = \textit{known sale quantity in Week 1})$$

2. Initial price is set at $60

$$P_1 = 60$$

3. Sum of sale quantities over all 15 weeks cannot exceed 2000

$$\sum_{i=1}^{15} Di < 2000$$

4. Price of week *i* must be greater than the price of week *i+1*

$$P_i \geq P_{i+1} \; for \; i \; in \; \{1,2,3 \ldots 14\}$$

5. Each week can only have 1 price out of $60, $54, $48 and $36

$$P_i - 60 * X_{i1} - 54 * X_{i2} - 48 * X_{i3} - 36 * X_{i4} = 0 \quad for \; i \; in \; \{1,2,3, \ldots ,15\}$$

6. To calculate sale quantity at each price, we created an array and calculated the sale quantity using the average demand jump with respect to the price of the item.

D1 = Initial Sale Quantity
D2 = 1.31*D1
D3 = 1.73*D1
D4 = 2.81*D1

$$D_i - D1 * X_{i1} - D2 * X_{i2} - D3 * X_{i3} - D4 * X_{i4} = 0 \quad for \; i \; in \; \{1,2,3, \ldots ,15\}$$

7. Dummy Variable Constraints (sum-product of dummy variables of each week must be equal to 1)

$$\sum_{j=1}^{4} X_{ij} = 1 \; for \; i \; in \; \{1,2,3, \ldots ,15\}$$

How to use the Optimization Model?

When you run the model, it will ask for an input: *'Enter First Demand Value at Week1'*. Just as shown in the Retailer Game, the Sale Quantity of the first week is automatically given. Therefore, with our model, we also created a function where D1 (Sale Quantity for Week 1) can be manually inputted.

When you run the model, the output generated is:

1. Optimal Revenue
2. Optimal Weekly pricing strategy

Using the output from our model, we played the Retailer Game with our optimized strategy and observed that in most cases, our optimal revenue was very close to the perfect foresight strategy, with a difference of <3%. Please see examples of the simulations below.

CONCLUSION

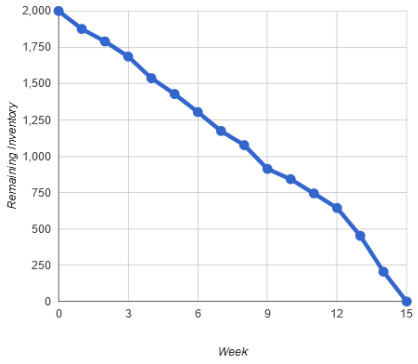
Overall, we have been able to create an optimization model which generates the optimal markdown strategy for a retailer based on the initial sale quantity, which is an indicator for demand.

SIMULATION

GAME SIMULATION

Restart Game | Compete Mode | *Game seed: 78945612*

Week	Price	Sales	Remaining Inventory
1	60	124	1876
2	60	86	1790
3	60	104	1686
4	60	149	1537
5	60	109	1428
6	60	124	1304
7	60	130	1174
8	60	98	1076
9	60	163	913
10	60	71	842
11	60	98	744
12	60	100	644
13	60	192	452
14	54	247	205
15	48	205	0



Maintain Price | 10% | 20% | 40%

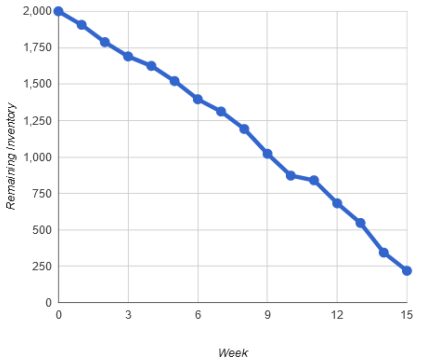
Your revenue: $116,058, Perfect foresight strategy: $116,853, Difference: 0.7%

MODEL RESULTS

```
###Optimal Strategy###
Revenue: 115907.76

Week 1: 60
Week 2: 60
Week 3: 60
Week 4: 60
Week 5: 60
Week 6: 60
Week 7: 60
Week 8: 60
Week 9: 60
Week 10: 60
Week 11: 60
Week 12: 60
Week 13: 60
Week 14: 54
Week 15: 48
```

Restart Game | Compete Mode | *Game seed: 5000*

Week	Price	Sales	Remaining Inventory
1	60	93	1907
2	60	119	1788
3	60	98	1690
4	60	65	1625
5	54	104	1521
6	54	125	1396
7	54	84	1312
8	54	120	1192
9	48	170	1022
10	48	150	872
11	48	32	840
12	48	158	682
13	48	135	547
14	48	203	344
15	48	125	219



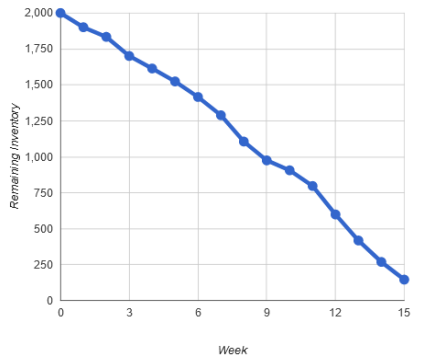
Maintain Price | 10% | 20% | 40%

Your revenue: $98,061, Perfect foresight strategy: $99,534, Difference: 1.5%

```
###Optimal Strategy###
Revenue: 103319.28

Week 1: 60
Week 2: 60
Week 3: 60
Week 4: 60
Week 5: 54
Week 6: 54
Week 7: 54
Week 8: 54
Week 9: 48
Week 10: 48
Week 11: 48
Week 12: 48
Week 13: 48
Week 14: 48
Week 15: 48
```

Restart Game | Compete Mode | *Game seed: 500*

Week	Price	Sales	Remaining Inventory
1	60	99	1901
2	60	67	1834
3	60	133	1701
4	60	87	1614
5	54	90	1524
6	54	108	1416
7	54	127	1289
8	54	183	1106
9	54	131	975
10	54	69	906
11	54	109	797
12	48	198	599
13	48	181	418
14	48	149	269
15	48	123	146



Maintain Price | 10% | 20% | 40%

Your revenue: $102,176, Perfect foresight strategy: $103,266, Difference: 1.1%

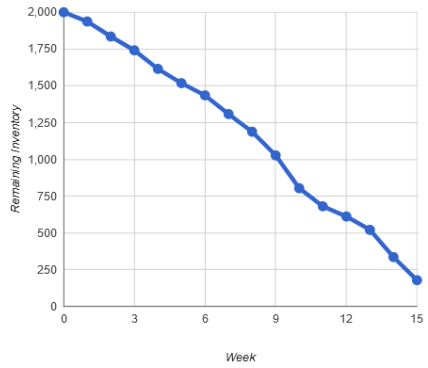
```
###Optimal Strategy###
Revenue: 106046.82

Week 1: 60
Week 2: 60
Week 3: 60
Week 4: 60
Week 5: 54
Week 6: 54
Week 7: 54
Week 8: 54
Week 9: 54
Week 10: 54
Week 11: 54
Week 12: 48
Week 13: 48
Week 14: 48
Week 15: 48
```

Restart Game | Compete Mode | *Game seed: 9999999*

Week	Price	Sales	Remaining Inventory
1	60	63	1937
2	54	102	1835
3	48	94	1741
4	48	126	1615
5	48	97	1518
6	48	83	1435
7	48	127	1308
8	48	120	1188
9	48	161	1027
10	36	223	804
11	36	123	681
12	36	69	612
13	36	91	521
14	36	185	336
15	36	157	179

Maintain Price | 10% | 20% | 40%

Your revenue: $83,075, Perfect foresight strategy: $86,130, Difference: 3.5%



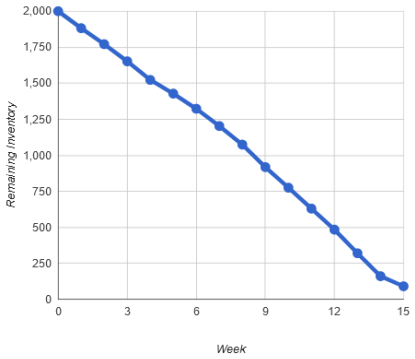

```
###Optimal Strategy###
Revenue: 83519.1


Week 1: 60
Week 2: 54
Week 3: 48
Week 4: 48
Week 5: 48
Week 6: 48
Week 7: 48
Week 8: 48
Week 9: 48
Week 10: 36
Week 11: 36
Week 12: 36
Week 13: 36
Week 14: 36
Week 15: 36
```

Restart Game | Compete Mode | *Game seed: 1234567*

Week	Price	Sales	Remaining Inventory
1	60	118	1882
2	60	111	1771
3	60	119	1652
4	60	129	1523
5	60	95	1428
6	60	105	1323
7	60	120	1203
8	60	129	1074
9	60	156	918
10	54	143	775
11	54	145	630
12	54	146	484
13	54	164	320
14	54	159	161
15	54	70	91

Maintain Price | 10% | 20% | 40%

Your revenue: $111,853, Perfect foresight strategy: $112,637, Difference: 0.7%



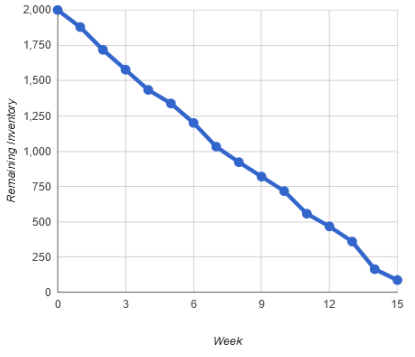

```
###Optimal Strategy###
Revenue: 113803.9200000


Week 1: 60
Week 2: 60
Week 3: 60
Week 4: 60
Week 5: 60
Week 6: 60
Week 7: 60
Week 8: 60
Week 9: 60
Week 10: 54
Week 11: 54
Week 12: 54
Week 13: 54
Week 14: 54
Week 15: 54
```

Restart Game | Compete Mode | *Game seed: 78979*

Week	Price	Sales	Remaining Inventory
1	60	121	1879
2	60	161	1718
3	60	141	1577
4	60	143	1434
5	60	96	1338
6	60	138	1200
7	60	168	1032
8	60	110	922
9	60	102	820
10	60	103	717
11	60	160	557
12	60	90	467
13	60	107	360
14	54	196	164
15	54	77	87

Maintain Price | 10% | 20% | 40%

Your revenue: $115,317, Perfect foresight strategy: $115,982, Difference: 0.6%




```
###Optimal Strategy###
Revenue: 114708.0


Week 1: 60
Week 2: 60
Week 3: 60
Week 4: 60
Week 5: 60
Week 6: 60
Week 7: 60
Week 8: 60
Week 9: 60
Week 10: 60
Week 11: 60
Week 12: 60
Week 13: 60
Week 14: 48
Week 15: 48
```