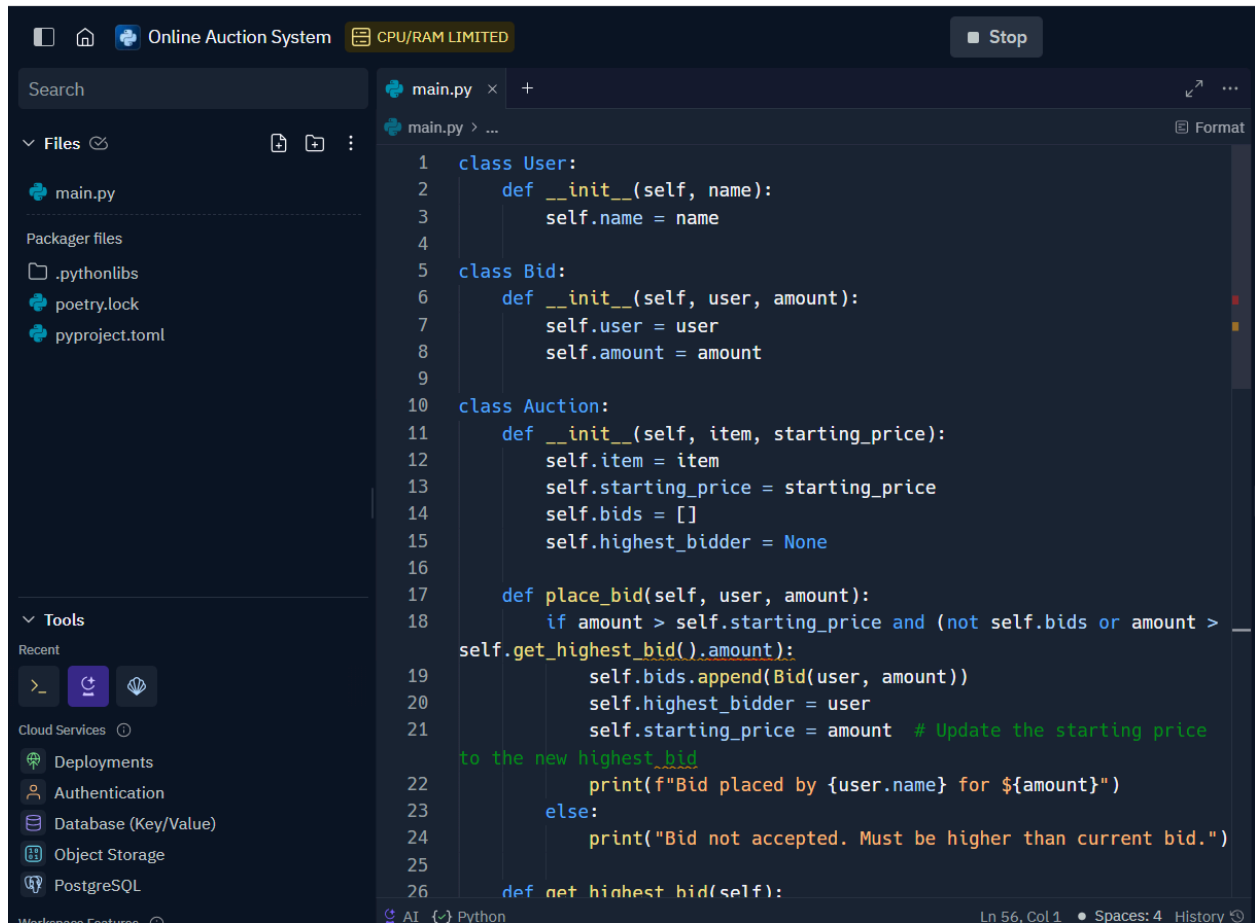


# Online Auction System

## Output:-



The screenshot displays a code editor window titled "Online Auction System" with a "CPU/RAM LIMITED" status bar and a "Stop" button. The left sidebar shows a file explorer with "main.py" and a "Tools" panel with various cloud services. The main editor area shows the following Python code:

```
1 class User:
2     def __init__(self, name):
3         self.name = name
4
5 class Bid:
6     def __init__(self, user, amount):
7         self.user = user
8         self.amount = amount
9
10 class Auction:
11     def __init__(self, item, starting_price):
12         self.item = item
13         self.starting_price = starting_price
14         self.bids = []
15         self.highest_bidder = None
16
17     def place_bid(self, user, amount):
18         if amount > self.starting_price and (not self.bids or amount >
19 self.get_highest_bid().amount):
20             self.bids.append(Bid(user, amount))
21             self.highest_bidder = user
22             self.starting_price = amount # Update the starting price
23             to the new highest bid
24             print(f"Bid placed by {user.name} for ${amount}")
25         else:
26             print("Bid not accepted. Must be higher than current bid.")
27
28     def get_highest_bid(self):
```

The code defines three classes: `User`, `Bid`, and `Auction`. The `Auction` class has methods for placing bids and retrieving the highest bid. The status bar at the bottom indicates "Ln 56, Col 1" and "Spaces: 4".

```
AI Console Shell +
Run Ask AI 44s on 22:06:06, 08/20
Auction for: Antique Vase
Starting Price: $100.0
No bids yet.
Enter your name (or 'exit' to quit): Lalithvarma
Enter your bid amount: 150
Bid placed by Lalithvarma for $150.0
Auction for: Antique Vase
Starting Price: $150.0
Highest Bid: $150.0 by Lalithvarma
Enter your name (or 'exit' to quit):
```

```
AI Console Shell +
Run Ask AI 27s on 22:07:04, 08/20
Auction for: Antique Vase
Starting Price: $100.0
No bids yet.
Enter your name (or 'exit' to quit): Lalithvarma
Enter your bid amount: 150
Bid placed by Lalithvarma for $150.0
Auction for: Antique Vase
Starting Price: $150.0
Highest Bid: $150.0 by Lalithvarma
Enter your name (or 'exit' to quit): Lalithvarma
Enter your bid amount: 95
Bid not accepted. Must be higher than current bid.
Auction for: Antique Vase
Starting Price: $150.0
Highest Bid: $150.0 by Lalithvarma
Enter your name (or 'exit' to quit):
```

```
Ask AI & search Ctrl K Invite PUBLIC Deploy ? LV
>_ Console x Shell +
Run Ask AI 39s on 22:07:04, 08/20 ✓

Auction for: Antique Vase
Starting Price: $100.0
No bids yet.
Enter your name (or 'exit' to quit): Lalithvarma
Enter your bid amount: 150
Bid placed by Lalithvarma for $150.0
Auction for: Antique Vase
Starting Price: $150.0
Highest Bid: $150.0 by Lalithvarma
Enter your name (or 'exit' to quit): Lalithvarma
Enter your bid amount: 95
Bid not accepted. Must be higher than current bid.
Auction for: Antique Vase
Starting Price: $150.0
Highest Bid: $150.0 by Lalithvarma
Enter your name (or 'exit' to quit): exit
```

```
Online Auction System CPU/RAM LIMITED Run
Search
Files
main.py
Packager files
pythonlibs
poetry.lock
pyproject.toml
Tools
Recent
Cloud Services
Deployments
Authentication
Database (Key/Value)
Object Storage
PostgreSQL
Workspace Features

main.py x +
main.py > ...
1 class User:
2     def __init__(self, name):
3         self.name = name
4
5 class Bid:
6     def __init__(self, user, amount):
7         self.user = user
8         self.amount = amount
9
10 class Auction:
11     def __init__(self, item, starting_price):
12         self.item = item
13         self.starting_price = starting_price
14         self.bids = []
15         self.highest_bidder = None
16
17     def place_bid(self, user, amount):
18         if amount > self.starting_price and (not self.bids or amount >
19         self.get_highest_bid().amount):
20             self.bids.append(Bid(user, amount))
21             self.highest_bidder = user
22             self.starting_price = amount # Update the starting price
23             to the new highest bid
24             print(f"Bid placed by {user.name} for ${amount}")
25         else:
26             print("Bid not accepted. Must be higher than current bid.")
27
28     def get_highest_bid(self):
29         # Implementation of get_highest_bid method
```

