

# Screen Shots :-

```
1 import random
2 import sys
3 import time
4
5
6 def get_valid_guess(lower, upper, previous_guesses): 1 usage
7     while True:
8         try:
9             guess = input(f"Enter your guess ({lower}-{upper}): ")
10            guess = int(guess)
11            if guess in previous_guesses:
12                print(f"You already guessed {guess}. Try a different number.")
13                time.sleep(1)
14                continue
15            if guess < lower or guess > upper:
16                print(f"Please enter a number between {lower} and {upper}.")
17                time.sleep(1)
18                continue
19            return guess
20        except ValueError:
21            print("Invalid input. Please enter a valid integer.")
22            time.sleep(1)
23
24
25 def analyze_guesses(guesses, secret, lower, upper): 1 usage
26     possible = set(range(lower, upper + 1))
27     for g in guesses:
28         if g < secret:
29             possible = set(x for x in possible if x > g)
30         elif g > secret:
31             possible = set(x for x in possible if x < g)
32     remaining = len(possible)
33     total = upper - lower + 1
34     prob = 1 - remaining / total
35     return max(0, min(1, prob))
36
37
38 def min_additional_tries(guesses, secret, lower, upper): 1 usage
39     # Calculate how many more tries would be needed assuming an optimal binary search on the reduced range
40     possible = set(range(lower, upper + 1))
41     for g in guesses:
42         if g < secret:
43             possible = set(x for x in possible if x > g)
44         elif g > secret:
45             possible = set(x for x in possible if x < g)
46     remaining = len(possible)
47     # Number of tries in worst-case binary search = ceil(log2(remaining))
48     import math
49     if remaining <= 1:
50         return 0
51     return math.ceil(math.log2(remaining))
52
53
```

```
51     return math.ceil(math.log2(remaining))  
52  
53  
54 def number_guessing_game(): 1 usage  
55     lower = 1  
56     upper = 100  
57     max_attempts = 7  
58     secret = random.randint(lower, upper)  
59     guesses = []  
60  
61     print(f"Welcome to the Number Guessing Game!")  
62     time.sleep(1)  
63     print(f"Guess the number between {lower} and {upper}. You have {max_attempts} tries.")  
64  
65     for attempt in range(1, max_attempts + 1):  
66         guess = get_valid_guess(lower, upper, guesses)  
67         guesses.append(guess)  
68  
69         if guess < secret:  
70             print("Try a higher number.")  
71         elif guess > secret:  
72             print("Try a lower number.")  
73         else:  
74             print(f"Congratulations! You guessed the number {secret} in {attempt} tries.")  
75             sys.exit(0)  
76             time.sleep(1)  
77         else:  
78             elif guess > secret:  
79                 print("Try a lower number.")  
80             else:  
81                 print(f"Congratulations! You guessed the number {secret} in {attempt} tries.")  
82                 sys.exit(0)  
83                 time.sleep(1)  
84             else:  
85                 print(f"Sorry, you've used all your tries. The number was {secret}. Try again!")  
86  
87  
88 probability = analyze_guesses(guesses, secret, lower, upper)  
89 print(f"Based on your answers, your probability of winning was approximately {probability:.2%}.")  
90  
91 more_tries_needed = min_additional_tries(guesses, secret, lower, upper)  
92 print(f"Based on your guesses so far,  
93      f"you could have found the secret number in about {more_tries_needed} more "  
94      f"optimally chosen guess{'es' if more_tries_needed != 1 else ''}.")  
95  
96  
97 if __name__ == "__main__":  
98     number_guessing_game()  
99  
100
```

The screenshot shows the PyCharm IDE interface. The top part is a terminal window titled "Run number\_gussing\_game". It displays the output of a Python script named "number\_gussing\_game.py". The script is a simple number guessing game. The user's input is shown in green, and the program's responses are in white. The bottom part is a "Project" tool window showing the file structure of the "PythonProject" directory. The "number\_gussing\_game.py" file is selected in the project tree. The status bar at the bottom right shows the current time as 10:50 PM and the date as 22-11-2025.

```
2 C:\Users\Hp\PycharmProjects\PythonProject\.venv\Scripts\python.exe C:\Users\Hp\PycharmProjects\PythonProject\number_gussing_game.py
3 Welcome to the Number Guessing Game!
4 Guess the number between 1 and 100. You have 7 tries.
5 Enter your guess (1-100): -1
6 Please enter a number between 1 and 100.
7 Enter your guess (1-100): 300
8 Please enter a number between 1 and 100.
9 Enter your guess (1-100): a
10 Invalid input. Please enter a valid integer.
11 Enter your guess (1-100): #
12 Invalid input. Please enter a valid integer.
13 Enter your guess (1-100): 2/5
14 Invalid input. Please enter a valid integer.
15 Enter your guess (1-100): 50
16 Try a lower number.
17 Enter your guess (1-100): 25
18 Try a lower number.
19 Enter your guess (1-100): 15
20 Try a higher number.
21 Enter your guess (1-100): 20
22 Try a lower number.
23 Enter your guess (1-100): 17
24 Congratulations! You guessed the number 17 in 5 tries.
25
26 Process finished with exit code 0
27
```

Project Tree (Left):

- PythonProject C:\Users\Hp\PycharmProjects\PythonProject
- |- venv library root
  - Assignment-1.py
  - Assignment-2.py
  - Assignment-3.py
  - Assignment-4.py
  - Chem.py
  - main.py
  - number\_gussing\_game.py
  - output.txt
  - PC001.py
  - PC002.py
  - PC003.py
  - PCM2.py
  - PCM3.py
  - PCM4.py
  - PCM5.py
  - PCM9.py
  - PCM10.py
  - sample.txt
  - Test.py
- > External Libraries
- > Scratches and Consoles

Terminal Output (Right):

```
2 C:\Users\Hp\PycharmProjects\PythonProject\.venv\Scripts\python.exe C:\Users\Hp\PycharmProjects\PythonProject\number_gussing_game.py
3 Welcome to the Number Guessing Game!
4 Guess the number between 1 and 100. You have 7 tries.
5 Enter your guess (1-100): 69
6 Congratulations! You guessed the number 69 in 1 tries.
7
8 Process finished with exit code 0
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
```

Status Bar (Bottom Right):

- 9:57
- CRLF
- UTF-8
- 4 spaces
- Python 3.13 (PythonProject)
- 10:50 PM
- ENG IN
- 22-11-2025

```
Run number_gussing_game x
2 C:\Users\Hp\PycharmProjects\PythonProject\.venv\Scripts\python.exe C:\Users\Hp\PycharmProjects\PythonProject\number_gu
3 Welcome to the Number Guessing Game!
4 Guess the number between 1 and 100. You have 7 tries.
5 Enter your guess (1-100): 50
6 Try a lower number.
7 Enter your guess (1-100): 2
8 Try a higher number.
9 Enter your guess (1-100): 1
10 Try a higher number.
11 Enter your guess (1-100): 18
12 Try a lower number.
13 Enter your guess (1-100): 14
14 Try a higher number.
15 Enter your guess (1-100): 99
16 Try a lower number.
17 Enter your guess (1-100): 50
18 You already guessed 50. Try a different number.
19 Enter your guess (1-100): 22
20 Try a lower number.
21 Sorry, you've used all your tries. The number was 16. Try again!
22 Based on your answers, your probability of winning was approximately 97.00%.
23 Based on your guesses so far, you could have found the secret number in about 2 more optimally chosen guesses.
24
25 Process finished with exit code 0
26
27
```