

Scraping **Booking.com** using Python



Challenges:

- Bypassing the bot checks
- Fetching as much data as needed
- Data Analysis



BENEFITS



- **Data Access:** Quickly gather large datasets from Booking.com.
- **Cost-Effective:** Saves money compared to manual data collection.
- **Speed:** Rapidly extract data for quick insights.
- **Competitive Advantage:** Gain insights into competitors' data.
- **Automation:** Automate data collection, saving time.



Sign up to:
<https://portal.netnut.io/>




Proxies ▾

Scrapers ▾

Datasets ▾

Pricing ▾

Resources ▾

 EN ▾

[Register](#)

[Login](#)



Pick your preferred Programming Language, Location & URL

The screenshot shows the Netnut Proxy Generator interface. At the top right, there is a notification bell, a user profile for 'Zia Ahmad' with a dropdown arrow, and a location selector showing the United States flag. Below this is the 'Proxy Generator' section with a terminal icon. It includes a 'Python' dropdown menu and a 'New Proxy' button. The main configuration area has three input fields: 'Proxy Port' (5959), 'Proxy server' (http://gw.ntnt.io), and 'Proxy user' (blurred). Below these are two dropdown menus: 'Datacenter' and 'United States' (with a US flag). A text input field contains the URL 'https://www.booking.com/'. At the bottom is a large orange 'Generate' button. Red arrows point from the top right towards the 'Python' dropdown, the 'United States' dropdown, and the URL input field.



After hitting **Generate** button
Copy this **Code & Paste** into the
Coding Environment



 Proxy
Generator

Python ▾

New Proxy

Here is a **basic CURL request** to our Datacenter network.
Copy and paste the code below into your cmd (win users) or
terminal (mac users).

Copy code 

```
# to use this example you must have  
'requests' installed: pip install requests  
import requests  
  
username = '  
password = '  
server = 'gw.ntnt.io'
```



- Go the Booking.com
- Select Destination & related fields
- Click Search button

Booking.com USD List your property **Hafiz Zia Ahmad**
Genius Level 1

[Stays](#) [Flights](#) [Car rentals](#) [Attractions](#) [TAXI](#) Airport taxis

[Home](#) > [France](#) > [Ile de France](#) > [Paris](#) > Search results

[Show on map](#)

Paris: 3,757 properties found

↓↑ Sort by: Our top picks

GAMBETTA résidences hôtel Paris centre Good 7.0
53 reviews

[20th arr., Paris](#) · [Show on map](#) · 3.5 km from centre · Metro access

Limited-time Deal

Superior Apartment
Entire apartment • 1 bedroom • 1 living room • 1 bathroom •
1 kitchen • 32m²
2 beds (1 double, 1 sofa bed)

Only 1 left at this price on our site

2 nights, 2 adults
~~US\$276~~ **US\$193** ⓘ
+US\$149 taxes and charges

[See availability](#)

Hotelf1 Paris Saint Ouen Marché Aux Review score 6.4
15,187 reviews

Filter by:

Your budget (per night)
US\$50 - US\$700+

Popular filters
 Free cancellation 2141



- **Copy the URL after search**
- **Go to Inspect**
- **Copy the classes of the targeted values**

A screenshot of a web browser showing search results on booking.com. The page lists three hotels in Paris: '20th arr., Paris', 'B Montmartre', and 'Le Katorze Hôtel'. The browser's developer tools are open on the right side, displaying the HTML structure of the page. The 'Elements' panel shows a tree view of the DOM, with several elements selected and highlighted in blue. These elements include various divs and spans that contain pricing and availability information, such as 'price-for-x-nights', 'price-and-discounted-price', and 'taxes-and-charges'. The selected elements have class attributes that are being targeted for extraction.

Importing Libraries

```
import requests
from bs4 import BeautifulSoup
```

Setting up the Proxy

```
username = 'ziaahmad-dc-US'
password = 'r8RNn8Qw6rDjcdM'
server = 'gw.ntnt.io'
port = '5959'
proxy = {'http': f'http://{username}:{password}@{server}:{port}'}

headers = {'User-Agent': 'Chrome/58.0.3029.110'}
url = "https://www.booking.com/searchresults.en-gb.html?ss=Paris"
response = requests.get(url, proxies=proxy, headers=headers)
```

Code for Data Mining

```
soup = BeautifulSoup(driver.page_source, 'html.parser')
post_cards = soup.find_all("div", "c624d7469d a0e60936ad a3214e5942 b0db0e8ada")
data = []
for i in post_cards:
    dct = {}
    try:
        dct['Hotel Name'] = i.find(class_="f6431b446c a15b38c233").get_text()
        dct['Price'] = i.find(class_="f6431b446c fbfd7c1165 e84eb96b1f").get_text()
        dct['Rating'] = i.find(class_="a3b8729ab1 d86cee9b25").get_text().split()[-1]
        dct['Room Type'] = i.find(class_="abf093bdfe e8f7c070a7").get_text()
        data.append(dct)
    except:
        print("Didn't find the Text")
pd.DataFrame(data)
```

Paste the classes that we copied at Step 5

Sample Output

Hotel Name	Price	Rating	Room Type
Royal Madeleine Hotel & Spa	US\$789	8.5	Classic Double Room
HotelF1 Paris Saint Ouen Marché Aux Puces	US\$165	6.4	Side-Car Room
Grand Hotel Dore	US\$412	8.2	Double Room
Hôtel Aiglon	US\$516	8.7	Cosy Double Room
Gardette Park Hotel	US\$535	8.1	Standard Double Room
Hotel Elysees Opera	US\$682	8.5	Superior Double Room
NH Paris Gare de l'Est	US\$585	7.7	Standard Room
La Belle Ville	US\$550	8.7	Double Room
Holiday Inn Express Paris-Canal De La Villette...	US\$461	8.1	Standard Room
Hotel Residence Foch	US\$568	8.2	Traditional Double or Twin Room
Grand Hotel Francais	US\$467	8.5	Superior Room
Hôtel Korner Etoile	US\$501	8.3	Twin Room
Best Western Bretagne Montparnasse	US\$389	8.3	Superior Queen Room - Non-Smoking
Hôtel Le Daum	US\$544	8.4	Comfort Double Room
Hotel de L'Esperance	US\$729	8.5	Junior Suite
Hotel Armoni Paris	US\$447	8.1	Standard Double Room
hotelF1 Paris Porte de Châtillon	US\$185	7.3	Side-Car Room
Le Katorze Hôtel	US\$624	8.3	Double Room
Hotel le 18 Paris	US\$360	7.7	Comfort Double Room
Kyriad Paris 18 - Porte de Clignancourt - Mont...	US\$331	8.1	Double Room



Summary

- Create an account
- Generate the script for proxy at [Netnut.io](https://netnut.io)
- Copy the desired URL
- Open any Python IDE
- Paste the script that we copied from netnut
- Edit the URL
- Mine the Data According to the need



Scraping Booking.com using Python

Hit Follow!



Save for later!



netnut.io