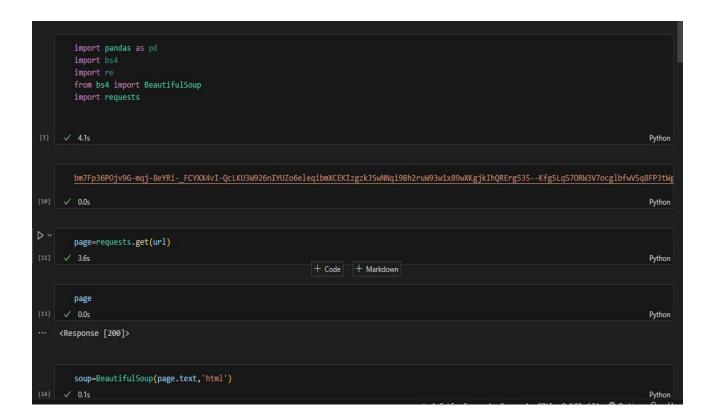WAP Scrape the following data from Amazon and save the data extracted in the CSV file.

Price, Amount, Description of a Mobile phone of any company.

```python
import pandas as pd
import bs4
import re
from bs4 import BeautifulSoup
import requests
```
[1] ✓ 4.1s   Python

bm7Fp36POjv9G-mqj-8eYRi-_FCYXX4vI-QcLKU3W926nIYUZo6eleqibmXCEKIzgzkJSwNNqi9Bh2ruW93w1x89wXKgjkIhQRErg535--Kfg5LqS7ORW3V7ocglbfwV5q8FP3tWg
[10] ✓ 0.0s   Python

```python
page=requests.get(url)
```
[11] ✓ 3.6s   Python

+ Code   + Markdown

```python
page
```
[13] ✓ 0.0s   Python

... <Response [200]>

```python
soup=BeautifulSoup(page.text,'html')
```
[14] ✓ 0.1s   Python

```python
soup
```
[15] ✓ 0.3s   Python

```
<!DOCTYPE html>
<html class="a-no-js" data-19ax5a9jf="dingo" lang="en-in"><!-- sp:feature:head-start -->
<head><script>var aPageStart = (new Date()).getTime();</script><meta charset="utf-8"/>
<!-- sp:end-feature:head-start -->
<!-- sp:feature:csm:head-open-part1 -->
<!-- sp:end-feature:csm:head-open-part1 -->
<!-- sp:feature:cs-optimization -->
<meta content="on" http-equiv="x-dns-prefetch-control"/>
<link crossorigin="" href="https://images-eu.ssl-images-amazon.com" rel="dns-prefetch"/>
<link crossorigin="" href="https://images-eu.ssl-images-amazon.com" rel="preconnect"/>
<link crossorigin="" href="https://m.media-amazon.com" rel="dns-prefetch"/>
<link crossorigin="" href="https://m.media-amazon.com" rel="preconnect"/>
<link crossorigin="" href="https://completion.amazon.com" rel="dns-prefetch"/>
<link crossorigin="" href="https://completion.amazon.com" rel="preconnect"/>
<!-- sp:end-feature:cs-optimization -->
<!-- sp:feature:csm:head-open-part2 -->
<!-- sp:end-feature:csm:head-open-part2 -->
<!-- sp:feature:aui-assets -->
<link href="https://m.media-amazon.com/images/I/11EIQ5IGqaL._RC|01e5ncglxyL.css,01lF2n-pPaL.css,41SwWPpN5yL.css,31+Z83i6adL.css,01IWMurvs8L.
<link as="script" crossorigin="anonymous" href="https://m.media-amazon.com/images/I/61xJcNKKLXL.js?AUIClients/AmazonUIjQuery" rel="preload">
<link as="script" crossorigin="anonymous" href="https://m.media-amazon.com/images/I/11Y+5x+kkTL._RC|51ExhNVPbdL.js,11yKORv-GTL.js,11GgN1+C7h
<link as="script" crossorigin="anonymous" href="https://m.media-amazon.com/images/I/51BqsgbDI7L.js?AUIClients/CardJsRuntimeBuzzCopyBuild" re
<script>
(function(b,a,c,d){if((b=b.AmazonUIPageJS||b.P)&&b.when&&b.register){c=[];for(a=a.currentScript;a;a=a.parentElement)a.id&&c.push(a.id);retur
c.name&&a.count(c.name,0===c.value?0:c.value||1);break;case "log_tag":a&&k(a.tag)&&c.tag&&(a.tag(c.tag),b=d.uex,a.isl&&k(b)&&b("at"));break;
...
<!--
```

```python
print(soup.prettify())
```
[16] ✓ 0.1s   Python

```
<!DOCTYPE html>
<html class="a-no-js" data-19ax5a9jf="dingo" lang="en-in">
 <!-- sp:feature:head-start -->
 <head>
  <script>
   var aPageStart = (new Date()).getTime();
  </script>
  <meta charset="utf-8"/>
  <!-- sp:end-feature:head-start -->
  <!-- sp:feature:csm:head-open-part1 -->
  <!-- sp:end-feature:csm:head-open-part1 -->
  <!-- sp:feature:cs-optimization -->
  <meta content="on" http-equiv="x-dns-prefetch-control"/>
  <link crossorigin="" href="https://images-eu.ssl-images-amazon.com" rel="dns-prefetch"/>
  <link crossorigin="" href="https://images-eu.ssl-images-amazon.com" rel="preconnect"/>
  <link crossorigin="" href="https://m.media-amazon.com" rel="dns-prefetch"/>
  <link crossorigin="" href="https://m.media-amazon.com" rel="preconnect"/>
  <link crossorigin="" href="https://completion.amazon.com" rel="dns-prefetch"/>
  <link crossorigin="" href="https://completion.amazon.com" rel="preconnect"/>
  <!-- sp:end-feature:cs-optimization -->
  <!-- sp:feature:csm:head-open-part2 -->
  <!-- sp:end-feature:csm:head-open-part2 -->
  <!-- sp:feature:aui-assets -->
  <link href="https://m.media-amazon.com/images/I/11EIQ5IGqaL._RC|01e5ncglxyL.css,01lF2n-pPaL.css,41SwWPpN5yL.css,31+Z83i6adL.css,01IWMurvs8
   <link as="script" crossorigin="anonymous" href="https://m.media-amazon.com/images/I/61xJcNKKLXL.js?AUIClients/AmazonUIjQuery" rel="preloa
   ...
```

Ln 1, Col 5   Spaces: 4   Spaces: 4   CRLF   Cell 23 of 24   Go Live   ⌵ {⌃

```python
soup.find_all('span')
```
[22] ✓ 0.0s   Python

```
[<span class="nav-sprite nav-logo-base"></span>,
 <span class="nav-sprite nav-logo-ext nav-progressive-content" id="logo-ext"></span>,
 <span class="nav-logo-locale">.in</span>,
 <span class="a-declarative nav-progressive-attribute" data-a-modal='{"width":375, "closeButton":"false","popoverLabel":"Choose your locatio
 <a class="nav-a nav-a-2 a-popover-trigger a-declarative nav-progressive-attribute" id="nav-global-location-popover-link" tabindex="0">
 <div class="nav-sprite nav-progressive-attribute" id="nav-packard-glow-loc-icon"></div>
 <div id="glow-ingress-block">
 <span class="nav-line-1 nav-progressive-content" id="glow-ingress-line1">
                Delivering to Mumbai 400001
            </span>
 <span class="nav-line-2 nav-progressive-content" id="glow-ingress-line2">
                Update location
            </span>
 </div>
 </a>
 </span>,
 <span class="nav-line-1 nav-progressive-content" id="glow-ingress-line1">
                Delivering to Mumbai 400001
            </span>,
 <span class="nav-line-2 nav-progressive-content" id="glow-ingress-line2">
                Update location
            </span>,
 <span class="nav-search-label nav-progressive-content" id="nav-search-label-id">Electronics</span>,
 <span aria-label="Go" class="nav-search-submit-text nav-sprite nav-progressive-attribute" id="nav-search-submit-text">
 <input class="nav-input nav-progressive-attribute" id="nav-search-submit-button" tabindex="0" type="submit" value="Go"/>
 ...
```

```python
title=soup.find_all("span",class_="a-size-large product-title-word-break")
```
[27] ✓ 0.0s

```python
Title=title[0].text.strip()
```
[28] ✓ 0.0s

Empty markdown cell, double-click or press enter to edit.

```python
Title
```
[29] ✓ 0.0s

```
'OnePlus 12 (Glacial White, 12GB RAM, 256GB Storage)'
```

```python
Price=soup.find_all('span',class_="a-price-whole")
```
[56] ✓ 0.0s

```python
price=Price[0].text.strip()
```
[57] ✓ 0.0s

```python
price
```
[59] ✓ 0.0s

```
'64,999.'
```

```python
Desc=soup.find_all('span',class_="a-size-base a-text-bold")
```
[60] ✓ 0.0s

```python
Desc
```
[61] ✓ 0.0s

```
[<span class="a-size-base a-text-bold" id="display-string"></span>,
 <span class="a-size-base a-text-bold">%cardName%</span>,
 <span class="a-size-base a-text-bold">Brand</span>,
 <span class="a-size-base a-text-bold">Operating System</span>,
 <span class="a-size-base a-text-bold">RAM Memory Installed Size</span>,
 <span class="a-size-base a-text-bold">CPU Model</span>,
 <span class="a-size-base a-text-bold">Memory Storage Capacity</span>]
```

```python
import pandas as pd
```
[62] ✓ 0.0s

```python
Data={'Title':[Title],
```

```python
import pandas as pd
```
[62] ✓ 0.0s   Python

```python
Data={'Title':[Title],
      'Price':[price]}
```
[64] ✓ 0.0s   Python

```python
df=pd.DataFrame(data=Data)
```
[65] ✓ 0.0s   Python

```python
df
```
[66] ✓ 0.0s   Python

	Title	Price
0	OnePlus 12 (Glacial White, 12GB RAM, 256GB Sto...	64,999.