```
from selenium import webdriver
from selenium.webdriver.edge.service import Service as EdgeService
from selenium.webdriver.edge.options import Options as EdgeOptions
from webdriver_manager.microsoft import EdgeChromiumDriverManager
from bs4 import BeautifulSoup
import pandas as pd
import time

# Setup Edge WebDriver
options = EdgeOptions()
options.add_argument("--start-maximized")
service = EdgeService(EdgeChromiumDriverManager().install())
driver = webdriver.Edge(service=service, options=options)

# Categories to scrape
categories = {
    "laptops": "laptop",
    "mobiles": "mobile"
}

all_data = []

# Loop through categories
for category_name, keyword in categories.items():
    print(f"\nScraping Category: {category_name.upper()}")
    category_data = []
    page = 1

    while len(category_data) < 500:
        print(f"Page {page}")
        url = f"https://www.amazon.in/s?k={keyword}&page={page}"
        driver.get(url)
        time.sleep(3)

        soup = BeautifulSoup(driver.page_source, "html.parser")
        results = soup.find_all("div", {"data-component-type": "s-search-
result"})

        for item in results:
            # Title
            title_tag = item.h2
            if title_tag:
                title = title_tag.text.strip()
            else:
                title = None

            # Price
            price_tag = item.find("span", class_="a-price-whole")
            if price_tag:
                price = price_tag.text.strip().replace(",", "")
            else:
                price = None

            # Rating
            rating_tag = item.find("span", class_="a-icon-alt")
            if rating_tag:
                rating = rating_tag.text.strip().split()[0]
            else:
                rating = None

            # Reviews
            reviews_tag = item.find("span", class_="a-size-base")
            if reviews_tag:
                reviews = reviews_tag.text.strip().replace(",", "")
```

```
            else:
                reviews = None

            # Brand
            if title:
                brand = title.split()[0]
            else:
                brand = None

            # Delivery
            delivery_tag = item.find("span", class_="a-color-base a-text-bold")
            if delivery_tag:
                delivery = delivery_tag.text.strip()
            else:
                delivery = None

            # Discount
            discount_tag = item.find("span", class_="a-letter-space")
            if discount_tag and discount_tag.find_next_sibling("span"):
                discount = discount_tag.find_next_sibling("span").text.strip()
            else:
                discount = None

            # Final check before saving
            if title and price and rating and reviews:
                category_data.append([
                    title, price, rating, reviews,
                    brand, delivery, discount, category_name
                ])
                print(title)

        page += 1
        if page > 30:
            break

    all_data.extend(category_data[:500])

# Close browser
driver.quit()
columns = ["Title", "Price", "Rating", "Reviews","Brand", "Delivery",
"Discount", "Category"]
```