連続する範囲のデータを得る方法

ファイル名: test100.xlsx

	A	B	C	D	E	F	G	H	I
1	A1	B1	C1	D1	E1	F1	G1	H1	I1
2	A2	B2	C2	D2	E2	F2	G2	H2	I2
3	A3	B3	C3	D3	E3	F3	G3	H3	I3
4	A4	B4	C4	D4	E4	F4	G4	H4	I4
5	A5	B5	C5	D5	E5	F5	G5	H5	I5
6	A6	B6	C6	D6	E6	F6	G6	H6	I6
7	A7	B7	C7	D7	E7	F7	G7	H7	I7
8	A8	B8	C8	D8	E8	F8	G8	H8	I8
9	A9	B9	C9	D9	E9	F9	G9	H9	I9
10	A10	B10	C10	D10	E10	F10	G10	H10	I10
11	A11	B11	C11	D11	E11	F11	G11	H11	I11
12	A12	B12	C12	D12	E12	F12	G12	H12	I12
13	A13	B13	C13	D13	E13	F13	G13	H13	I13
14	A14	B14	C14	D14	E14	F14	G14	H14	I14
15	A15	B15	C15	D15	E15	F15	G15	H15	I15

①for文により1つずつセルを得る

```
import openpyxl as excel

# ワークブックを開いてシートを得る --- (*1)
book = excel.load_workbook('/Users/ test100.xlsx'
sheet = book.active

# 連続でセルの値を得て表示 --- (*2)
for y in range(2, 5):
    r = []
    for x in range(2, 5):
        v = sheet.cell(row=y, column=x).value
        r.append(v)
    print(r)
```

行y、列xの値を取得してvに入れる

appned(v)　配列vに値を追加

②左上と右下を指定して得る

```
import openpyxl as excel

# ワークブックを開いてシートを取り出す
book = excel.load_workbook('/Users/ test100.xlsx'
sheet = book.active

# 連続でセルの値を得る
for row in sheet["B2":"D4"]:
    r = []
    for cell in row:
        r.append(cell.value)
    print(r)
```

←行毎にセルの値をrに追加

```
import openpyxl as excel

# ワークブックを開いてシートを得る
book = excel.load_workbook('/Users/ test100.xlsx'
sheet = book.active

# 連続でセルの値を得る
for row in sheet["B2":"D4"]:
    print([c.value for c in row])
```

③item_rowsで繰り返し得る

```
import openpyxl as excel

# ワークブックを開いてシートを取り出す
book = excel.load_workbook('/Users/ test100.xlsx'
sheet = book.active

# イテレータを取得する --- (*1)
it = sheet.iter_rows(
        min_row=2, min_col=2,
        max_row=4, max_col=4)
# for文で繰り返し値を得る --- (*2)
for row in it:
    r = []
    for cell in row:
        r.append(cell.value)
    print(r)
```

実行結果 ①、②及び③何も同じ

```
['B2', 'C2', 'D2']
['B3', 'C3', 'D3']
['B4', 'C4', 'D4']
```

	A	B	C	D	E	F
1	11月 売上データ一覧					
2	日付	購入者	品目	個数	値段	小計
3	11月3日	Aさん	バナナ	3	1,000	3,000
4	11月5日	Bさん	リンゴ	2	1,500	3,000
5	11月5日	Cさん	イチゴ	2	2,000	4,000
6	11月7日	Dさん	バナナ	8	1,000	8,000
7	11月15日	Bさん	リンゴ	2	1,500	3,000
8	11月16日	Dさん	イチゴ	5	2,000	10,000
9	11月20日	Eさん	マンゴー	3	1,900	5,700
10						

← =D3*E3の計算式が入っている

```python
import openpyxl as excel

# 売上データのブックを開いてシートを取り出す
book = excel.load_workbook("/Users/ uriage.xlsx,')
sheet = book.active

# A3からF9のセルを取り出す --- (*1)
rows = sheet["A3":"F9"]
for row in rows:
    # セルの値をリストとして得る --- (*2)
    values = [cell.value for cell in row]
    # リストを表示する --- (*3)
    print(values)
```

セルそのまま表示

実行結果

```
[datetime.datetime(2022, 11, 3, 0, 0), 'Aさん', 'バナナ', 3, 1000, '=D3*E3']
[datetime.datetime(2022, 11, 5, 0, 0), 'Bさん', 'リンゴ', 2, 1500, '=D4*E4']
[datetime.datetime(2022, 11, 5, 0, 0), 'Cさん', 'イチゴ', 2, 2000, '=D5*E5']
[datetime.datetime(2022, 11, 7, 0, 0), 'Dさん', 'バナナ', 8, 1000, '=D6*E6']
[datetime.datetime(2022, 11, 15, 0, 0), 'Bさん', 'リンゴ', 2, 1500, '=D7*E7']
[datetime.datetime(2022, 11, 16, 0, 0), 'Dさん', 'イチゴ', 5, 2000, '=D8*E8']
[datetime.datetime(2022, 11, 20, 0, 0), 'Eさん', 'マンゴー', 3, 1900, '=D9*E9']
```

```python
import openpyxl as excel

# 売上データのブックを開いてシートを取り出す
book = excel.load_workbook("/Users/ uriage.xlsx,', data_only=True)
sheet = book.active

# A3からF9のセルを取り出す --- (*1)
rows = sheet["A3":"F9"]
for row in rows:
    # セルの値をリストとして得る --- (*2)
    values = [cell.value for cell in row]
    # リストを表示する --- (*3)
    print(values)
```

計算結果が表示

実行結果

```
[datetime.datetime(2022, 11, 3, 0, 0), 'Aさん', 'バナナ', 3, 1000, 3000]
[datetime.datetime(2022, 11, 5, 0, 0), 'Bさん', 'リンゴ', 2, 1500, 3000]
[datetime.datetime(2022, 11, 5, 0, 0), 'Cさん', 'イチゴ', 2, 2000, 4000]
[datetime.datetime(2022, 11, 7, 0, 0), 'Dさん', 'バナナ', 8, 1000, 8000]
[datetime.datetime(2022, 11, 15, 0, 0), 'Bさん', 'リンゴ', 2, 1500, 3000]
[datetime.datetime(2022, 11, 16, 0, 0), 'Dさん', 'イチゴ', 5, 2000, 10000]
[datetime.datetime(2022, 11, 20, 0, 0), 'Eさん', 'マンゴー', 3, 1900, 5700]
```

	A	B	C	D	E	F
1	11月 売上データ一覧					
2	日付	購入者	品目	個数	値段	小計
3	11月3日	Aさん	バナナ	3	1,000	3,000
4	11月5日	Bさん	リンゴ	2	1,500	3,000
5	11月5日	Cさん	イチゴ	2	2,000	4,000
6	11月7日	Dさん	バナナ	8	1,000	8,000
7	11月15日	Bさん	リンゴ	2	1,500	3,000
8	11月16日	Dさん	イチゴ	5	2,000	10,000
9	11月20日	Eさん	マンゴー	3	1,900	5,700
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20						
21						
22						

```python
import openpyxl as excel

# 売上データのブックを開いてシートを取り出す
book = excel.load_workbook(
  "/Users/uriage.xlsx", data_only=True)
sheet = book.active

# A3からF999(データの適当な範囲)を取り出す --- (*1)
rows = sheet["A3":"F999"]
for row in rows:
   # セルの値をリストとして得る --- (*2)
   values = [cell.value for cell in row]
   # 空白セルであれば読み取りを終わる --- (*3)
   if values[0] is None: break
   # リストを表示する
   print(values)
```

←適当な最下行までの範囲を指定

←空セルはNone

実行結果

```
[datetime.datetime(2022, 11, 3, 0, 0), 'Aさん', 'バナナ', 3, 1000, 3000]
[datetime.datetime(2022, 11, 5, 0, 0), 'Bさん', 'リンゴ', 2, 1500, 3000]
[datetime.datetime(2022, 11, 5, 0, 0), 'Cさん', 'イチゴ', 2, 2000, 4000]
[datetime.datetime(2022, 11, 7, 0, 0), 'Dさん', 'バナナ', 8, 1000, 8000]
[datetime.datetime(2022, 11, 15, 0, 0), 'Bさん', 'リンゴ', 2, 1500, 3000]
[datetime.datetime(2022, 11, 16, 0, 0), 'Dさん', 'イチゴ', 5, 2000, 10000]
[datetime.datetime(2022, 11, 20, 0, 0), 'Eさん', 'マンゴー', 3, 1900, 5700]
```

```python
import openpyxl as excel

# ワークブックを開いてシートを得る --- (*1)
book = excel.load_workbook('/Users/ uriage_keisen.xlsx,')
sheet = book.active

# 連続でセルの値を得て表示 --- (*2)
for y in range(3, sheet.max_row):
   r = []
   for x in range(1, 6):
      v = sheet.cell(row=y, column=x).value
      r.append(v)
   print(r)
```

罫線がある場合、最下行を指定すると罫線部分の空セルも取得してしまう

実行結果

```
[datetime.datetime(2022, 11, 3, 0, 0), 'Aさん', 'バナナ', 3, 1000]
[datetime.datetime(2022, 11, 5, 0, 0), 'Bさん', 'リンゴ', 2, 1500]
[datetime.datetime(2022, 11, 5, 0, 0), 'Cさん', 'イチゴ', 2, 2000]
[datetime.datetime(2022, 11, 7, 0, 0), 'Dさん', 'バナナ', 8, 1000]
[datetime.datetime(2022, 11, 15, 0, 0), 'Bさん', 'リンゴ', 2, 1500]
[datetime.datetime(2022, 11, 16, 0, 0), 'Dさん', 'イチゴ', 5, 2000]
[datetime.datetime(2022, 11, 20, 0, 0), 'Eさん', 'マンゴー', 3, 1900]
[None, None, None, None, None]
[None, None, None, None, None]
[None, None, None, None, None]
[None, None, None, None, None]
[None, None, None, None, None]
[None, None, None, None, None]
[None, None, None, None, None]
[None, None, None, None, None]
[None, None, None, None, None]
[None, None, None, None, None]
[None, None, None, None, None]
```