



国立大学法人奈良女子大学

Nara Women's University

プログラミング講座

2016年12月22日

奈良女子大学

教授 駒谷 昇一

skomaya@cc.nara-wu.ac.jp



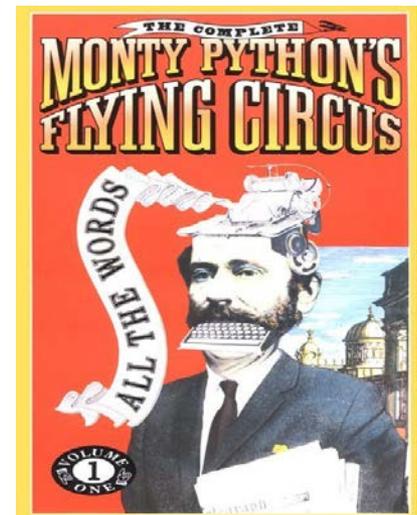
記念館(重要文化財)



プログラミング講座の概要

2

- 社会の抱える問題をコンピュータで解決するためのプログラムを作成します
- プログラムを作ることをプログラミングといいます
- プログラムは、目的により作り方(言語)が違います
- 今回は、Python(パイソン)という言語を使います
初心者を使いやすく、実際の仕事でも使われています



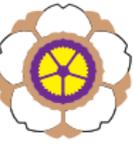


プログラミング講座のスケジュール

3

□ 1日目

- 13:00～ Pythonのインストール、Path変数
コマンドプロンプトでPythonの命令を実行
プログラムをファイルに書いて動かす
IDLEの使い方
プログラミングの基礎
- 15:00～ バーコードの仕組み
ビジネスモデルを考える
データ設計
マスターデータを作成
機能を考える
ファイルを読み込むプログラムの作成



プログラミング講座のスケジュール

4

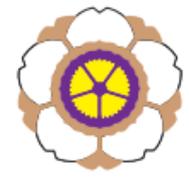
□ 2日目

- 午前 商品の販売を行うPOSレジの作成
- 午後 商品の販売状況を確認するプログラムの作成
プログラムの発表会(くじ引き)

ルール

5

- プログラムの作成はチームで相談して行うこと！
- プログラムを改造するときは、何のために、どこを改造するのかをチームで確認しながら改造する！
- 個人的に色々試してみるのもいいけれど、その結果をチームで共有すること！
- 私が作っているプログラムの内容を突然、誰かに訊きますが、必ず答えられなければなりません！
答えられないと罰ゲームがあります



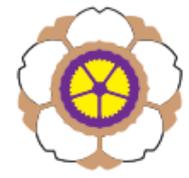
6

Pythonのインストール

Pythonのインストール

7

- Pythonのダウンロード
 - www.python.jp/ から、
『Python 3.5.1』の『ダウンロード』を選択
 - 『Files』の『[Windows x86-64 executable installer](#)』を
選択してダウンロードする(約30MBの容量)
- Pythonのインストール
 - ダウンロードしたファイルをデスクトップに置いて、
ダブルクリックで実行する
 - Path変数の設定もを入れておくと設定してくれる



国立大学法人奈良女子大学
Nara Women's University

8

Path変数

コマンドプロンプトとは

9

- WindowsのOSにあるプログラムやコマンド(命令)を実行するためのソフトウェア(シェルという)をコマンドプロンプトといいます。
- Windowsの左下の  でマウスの右クリックすると表示されるメニューからコマンドプロンプトを選び、実行することができます。
- 『dir』と入力するとファイルやフォルダが表示されます。ちなみに、dirはディレクトリの略です。
- 『set』と入力すると環境変数の値が表示されます。
- Pathという変数には何が書かれていますか？

Pathの意味

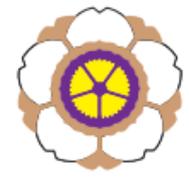
10

- コマンドプロンプトで『komaya』と入力すると、

```
'komaya' は、内部コマンドまたは外部コマンド、  
操作可能なプログラムまたはバッチ ファイルとして認識されていません。
```

と表示されます。Windowsにはkomayaというコマンドは無いということです。

- コマンドは、プログラムのファイルで、Path変数に書かれているフォルダを探しに行き、無いと上のメッセージを表示します。
- Pythonのプログラムを実行するためには、インストールされたフォルダをPath変数に追加する必要があります。



コマンドプロンプトでPythonの命令を実行

数値計算

12

- ではコマンドプロンプトで『python』と入力してみよう
- Pythonが動き、『>>>』が表示されます。
これを『プロンプト』といい、Pythonの命令を受け付ける準備ができたことを示しています。
- 半角文字で、『1+2』と入力して、改行してみましよう。
- 『1/3』は？
- 『7%3』は？
- 『2*10』、『2**10』は？
- 『2>10』、『2<10』、『2<=2』、『2==2』は？
- 終了は、Ctrlキーを押しながらZキーを押して改行

変数

13

- 『price=100』の後、『price』や『price*2』はどう？
- 『name='tanaka'』の後、『name』は？
『name*3』は？
- 『weight=[50,60,70]』の後、『weight』は？
『weight[2]』は？
『weight[1]』は？、『weight[0]』は？
- 『fluits=['apple', 'orange']』の後『fluits』は？
さらに、『fluits[1]』は？『fluits[0]』は？
『fluits[2]』は？

補足 コマンドインタープリター

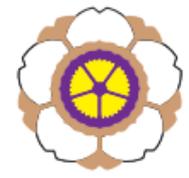
14

- Pythonでない他の言語では、プログラムを書いたファイル(ソースプログラム)を作成し、コンパイルという作業を行い、コンピュータが実行可能なファイル(オブジェクトファイル)を作成して、初めて実行が可能となります。
- しかし Pythonというプログラミング言語では、コマンドを入力するとすぐに実行してくれます。これをインタープリター型のプログラミング言語といいます。
- コマンドを受け付けて、実行してくれるプログラムのことをコマンドインタープリターといいます。

まとめ

15

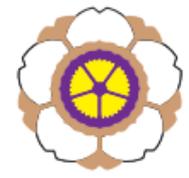
- 四則演算、あまりなどの計算ができます。
- データには、整数、小数点のある数値、文字列があり、変数に値を入れて記憶させることができます。
- 変数には、配列といって複数のデータを1つにまとめて記憶させることができ、[0]や[1]で個々の内容を知ることができます。
- 判別をすることができ、正しい場合は『True』、誤っている場合は『False』と表示されます。



16

プログラムをファイルに書いて動かす

- プログラムのファイルは、テキストドキュメントで、『メモ帳』や『Edit with IDLE』などで作成します。
- ファイル名は、『〇〇.py』のように最後に『.py』を付けます。ファイル名の〇の部分は英数字や記号です。
- 『Edit with IDLE』だと、Pythonのプログラムを作りやすいようにカラーで表示されたり、実行もできるので便利です。
- 初めは、マウスの右クリックで、新規作成、テキストドキュメントを選び、ファイル名を『〇〇.txt』ではなく、『〇〇.py』にして作成します。

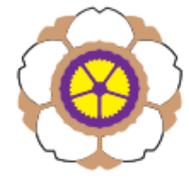


IDLEの使い方

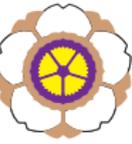
IDLEとは、Pythonの統合開発環境

19

- IDLEとは、Integrated DeveLopment Environmentの略
- Pythonのプログラムファイルを開く際に、右クリックで選択することができる。
- プログラムを作成したり、実行したりするF5キーで実行することができる
- コメント(#で始まる行)は赤色で表示される。
- 文字列は緑色で表示される。
- 制御構造(for,if,whileなど)はオレンジ色で表示される。
- インデントも自動的につけてくれる。



プログラミングの基礎



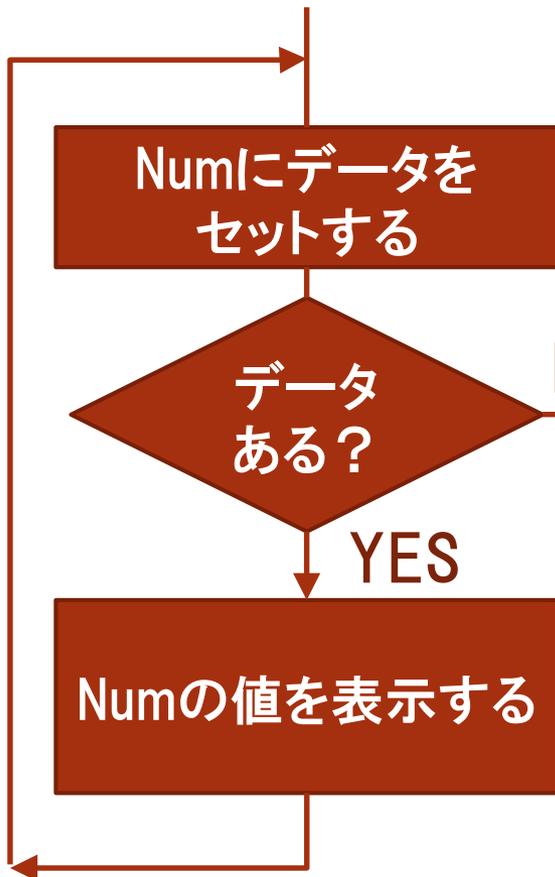
繰り返し(for)

21

□ ○○を繰り返し実行する

```
for num in [ 11, 12, 13, 14 ]:  
    print( num )
```

この4つの空白をインデントといいます
空白は必須です



繰り返し(for)

22

- 以下のプログラムを動かしてみる

```
for i in [ apple, orange, grapes]:
    print( i )
```

NameError: name 'apple' is not defined

というエラーが表示される。

Appleを変数とみなしていて、
変数の値が定義(define)されていない、
というよというエラーメッセージ。

繰り返し(for)

23

- 以下のプログラムを動かしてみる

```
apple = 101  
orange = 102  
grapes = 103  
for num in [ apple, orange, grapes ]:  
    print( num )
```

- やりたかったことはこんな感じでできるよ

```
for num in [ 'apple', 'orange', 'grapes' ]:  
    print( num )
```

繰り返し(for)

24

- 以下のプログラムはどうかな？

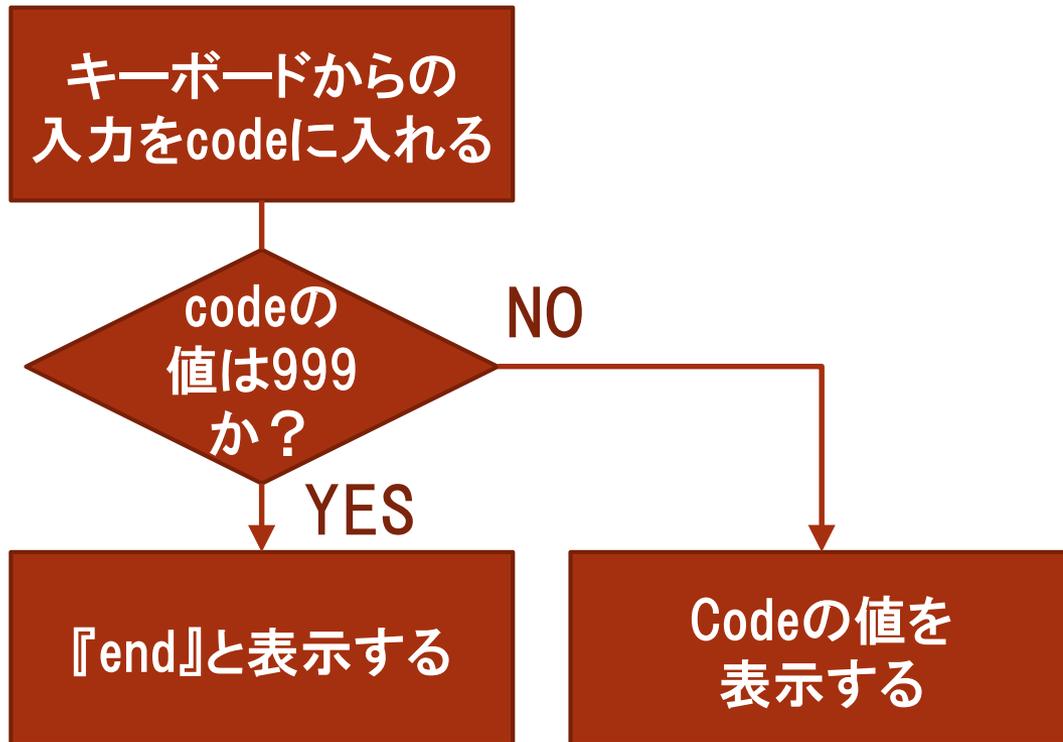
```
code = input()  
print( code )  
for i in code:  
    print( i )
```



分岐(if)

25

- もし〇〇だったら、〇〇を実行する



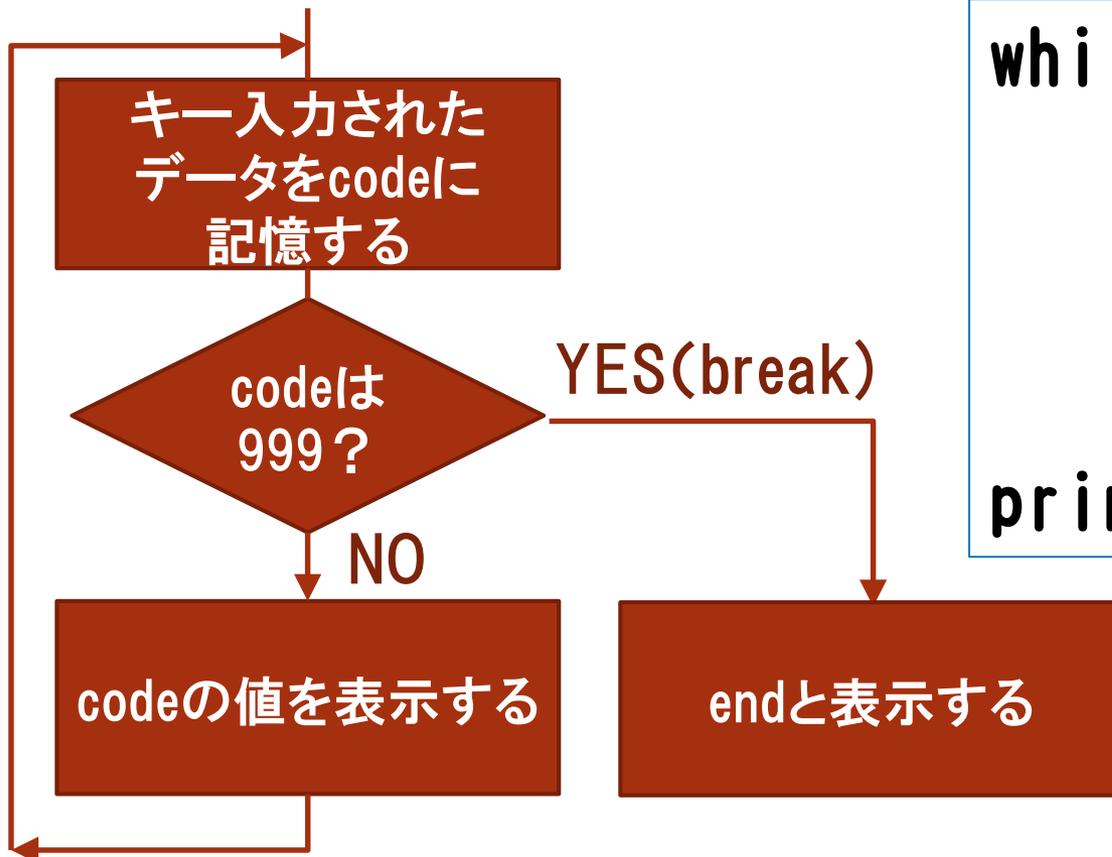
```
code = input()
if code == '999':
    print( 'end' )
else:
    print( code )
```



無限ループ(while)

26

- 繰り返し実行し、条件を満たしたらループを抜ける

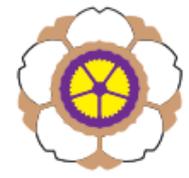


```
while True:  
    code = input()  
    if code == '999':  
        break  
    print( code )  
print( 'end' )
```

プログラム作成での注意

27

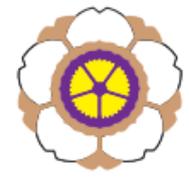
- プログラムは半角英数字で書きます。
- 全角文字が混ざっているとエラーになります。
- 文字列を表す場合は、『 』を使います。
- 行の先頭に『#』が書かれているとコメントになります
プログラムの内容を忘れないために、コメントを書くようにしましょう。日本語も使えます。
コメントは赤色で表示されます。
- 変数名は意味が分かる名前にします。
カウンタの場合は、『i』や『j』という1文字の変数を用いることが多いです。



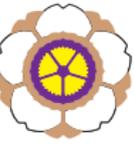
国立大学法人奈良女子大学
Nara Women's University

28

休憩



バーコードの仕組み



バーコードリーダー

30

- バーコードリーダーをパソコンに接続(USB接続)すると、キーボードとして認識され、バーコードの数字がキーボードから入力されたのと同じ動作をします。



バーコードの仕組み

31

- バーコードは下の数値を線の太さの違いで表現したものです。



①	国コード
②	企業(メーカー)コード
③	商品アイテムコード
④	チェックデジット

- バーコード(JAN)は4つの部分から構成されています。
 - 国コード 日本 45と49 クーポン 99 店舗内 200~299
 - 企業コード 5または7桁 ※過去は5桁でしたが、現在は7桁
 - 商品コード 5または3桁
 - チェックデジット 1桁 (読み取り誤りを発見するためのデータ)
- バーコードには、商品名、金額などは書かれていません。

参考 チェックデジットの計算方法

32

モジュラス10ウェイト3の計算方法は以下の通り。

(1)データキャラクタの最も右側にあるキャラクタを奇数とし、左へ順番に全てのデータキャラクタを奇数位置と偶数位置に分類する。

【例】JANコードデータが「4912345」の場合。

奇数位置:4,1,3,5 偶数位置:9,2,4

(2)奇数位置にあるキャラクタを合計し、その結果を3倍する。

$$(4+1+3+5) \times 3 = 39$$

(3)偶数位置のキャラクタを合計する。

$$9+2+4=15$$

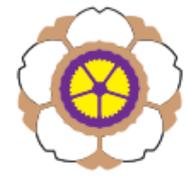
(4)奇数位置の結果(2)と偶数位置の結果(3)を合計する。

$$39+15=54$$

(5)「10」から(4)の答の1の位の数字を引いた数字がチェックデジットになる。

$$10-4=6 \quad \text{この場合のチェックデジットは「6」で、データは「49123456」となる。}$$

なお、(4)の答の1桁目が「0」の場合は、チェックデジットは「0」になる。



ビジネスモデルを考える

コンビニのレジのキー

34

女性 男性

12	12	小学生
19	19	中高学生
29	29	20歳代
49	49	30,40歳代
50	50	50歳以上

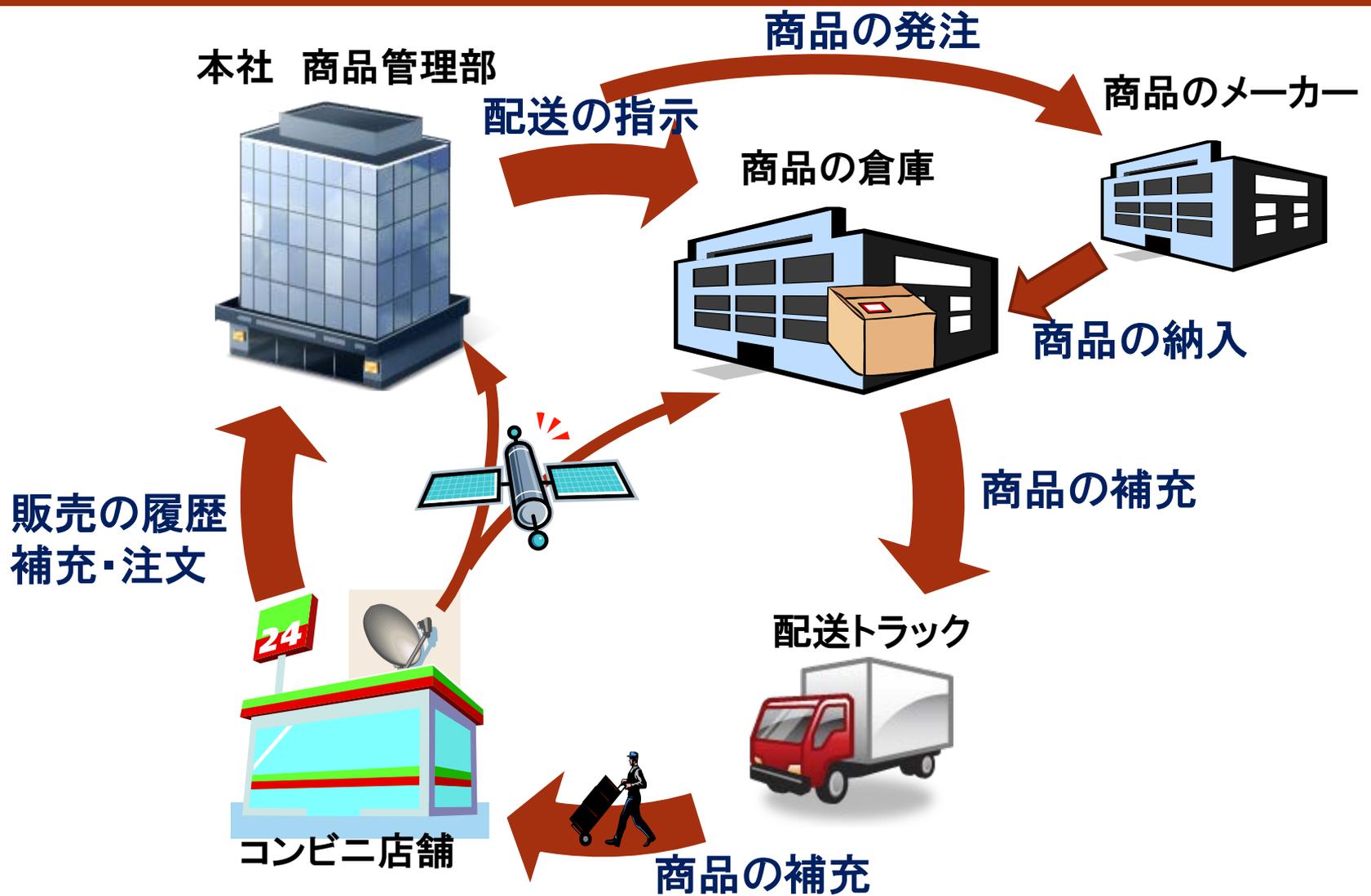
□ これは、年齢と性別のボタンです。

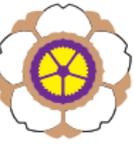
□ コンビニで購入した人の年齢と性別をこのボタンで入力しています。

□ 『どの商品』が、『いつ』、『どこの店』で、どのような『年齢層』、『性別』の人に、『どの商品と一緒に』売れたのか、のデータを得ているのですが、何のためにデータを集めているのでしょうか？



コンビニの配送の仕組み

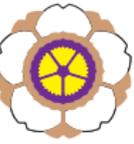




コンビニにある商品って同じ？

36

- コンビニの店舗によって、売れる商品と数と時間帯は異なります。例えば、お弁当の場合、
 - 高校や大学の近く(学生街)
 - オフィス街
 - 住宅街
 - 繁華街、観光地(海、山、花火大会会場近く)など
- お弁当の早い時間帯での売り切れ(機会損失)と、売れ残り(無駄)を無くすためには、どうしたらよいでしょう？



困ってます

37

- 道の駅で、地元で採れた野菜を売っています。
- しかし、売り切れていたり、売れ残ったりしているのが分からないので困っています。
- 売り切れていたら、商品を補充したいし、売れ残っていたら値段を下げたりしたいのです。
- どうしたらいいですか？



売主、販売店、お客様

38





それぞれの要望

39

□ 売主の要望



- 商品の在庫状況を知りたい
- 売れ残っていたら、値段を下げたい
- 前日の売り上げ内容を知りたい

□ 販売店の要望



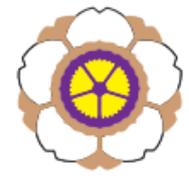
- 値段の入力ミス無くしたい、会計処理を早くしたい
- 毎日、売主ごとの売り上げ金額の集計を簡単にできるようにしてほしい



販売店で行うこと

40

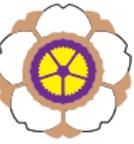
- 初期設定
 - ▣ 売主の情報をシステムへ登録
 - ▣ 商品の情報をシステムへ登録
- お客様ごとの処理
 - ▣ バーコードでレジを行う
 - ▣ 明細書(レシート)を渡し、代金を受け取る
- 毎日の開店前と閉店後の処理
 - ▣ 開店前 売主から商品の入荷があり、在庫数を増やす
 - ▣ 閉店後 売主ごとのその日の売り上げを集計する
集計結果をもとに翌日に代金を支払う



国立大学法人奈良女子大学
Nara Women's University

41

データ設計



マスターデータとログデータ

42

□ マスターデータとは

- データを作成したら、あまり変更がないもの
追加や削除は少なく、多くは参照されるのみ
- 学生番号、口座番号、IDとパスワード、名簿

□ ログデータとは

- 頻繁にデータの更新(追加や削除)が行われるもの
- お小遣い帳、出席簿、LINEのデータ、視聴履歴



マスターデータとログデータ

43

□ 初期設定

- 売主の情報をシステムへ登録
- 商品の情報をシステムへ登録

} マスターデータ

□ お客様ごとの処理

- バーコードでレジを行う
- 明細書(レシート)を渡し、代金を受け取る

} ログデータ
(履歴)

□ 毎日の閉店後の処理

- 売主ごとのその日の売り上げを集計する
集計結果をもとに翌日に代金を支払う

マスターデータ

44

- 売主マスターデータ
 - ▣ 売主コード(バーコード)、氏名、住所、電話番号
- 商品マスターデータ
 - ▣ バーコード(商品コード)、品名、売値、 他には？
 - ▣ 同じ商品でも、売主により、バーコードは異なります。
 - ▣ 日により、同じ商品でも売値は違います。
 - ▣ 同じ日でも、売れ残っていたら、売値を下げます。

マスターデータのデータ項目には何が必要か、
考えてみましょう。

レジの業務

45

- バーコードを読み取ると商品名と価格が表示されます。
- 会計ボタンを押すと合計金額が表示されます。
- 商品が売れ残っていた場合は、
売主からの要望で価格を下げることもあります。

- 使うデータは何？
 - バーコード、商品名、価格、他には？

ログデータには、どのような情報を記録すればいい？

開店前の処理

46

- 売主が商品を店舗に持ってきます。
- 商品にバーコードを貼り付けます。
- 商品の価格は日によって異なります。
- 季節により新しい商品を売り始めることもあります。

- 使うデータは何？
 - ▣ 商品の価格の変更
 - ▣ 入荷数？、他には？

閉店後の処理

47

- 売主ごとに、その日の商品の販売履歴のデータを集計し、売主に支払う代金の合計の計算を行います。

- 使うデータは何？
 - 商品名、売上数、販売価格、売り上げ金額
 - 合計金額、他には？

ログデータ

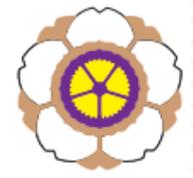
48

□ 販売履歴データ

- 販売した商品のバーコード
- 販売の年月日時分秒
- 売値 他には？

- 集計がし易いように履歴データを設計することが重要！

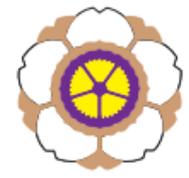
ログデータのデータ項目には何が必要か、
考えてみましょう。



国立大学法人奈良女子大学
Nara Women's University

49

休憩



マスターデータを作成

マスターデータを作成する

51

- ファイルは、テキストファイルで作ります。
- ファイル名は、『〇〇.csv』とします。
- 1つの商品や売主は、1行で表します
- データは、『,』で区切ります。
- これらのデータ形式をCSV形式といいます。
- CSV形式のデータは、EXCELでも作成できます。
EXCELで保存する際に、CSV(カンマ区切り)とすると『,』で区切られたCSV形式のデータが作成できます。

CSV形式のデータ

52

□ 例えば、右のEXCELデータをCSV形式で保存すると下のよう
なデータができる

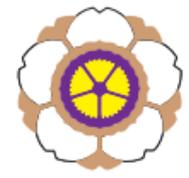
郵便番号	県	市	町名	県	市	町名
6308296	ナラケン	ナラシ	ゴトウチョウ	奈良県	奈良市	後藤町
6308297	ナラケン	ナラシ	キタワハタチョウ	奈良県	奈良市	北川端町
6308301	ナラケン	ナラシ	タカバタケチョウ	奈良県	奈良市	高畑町
6308302	ナラケン	ナラシ	ヒヤクゴウジチョウ	奈良県	奈良市	白毫寺町
6308303	ナラケン	ナラシ	ミナミキテラチョウ	奈良県	奈良市	南紀寺町
6308304	ナラケン	ナラシ	ミナミカイノヅカチョウ	奈良県	奈良市	南肘塚町
6308305	ナラケン	ナラシ	ヒガシキテラチョウ	奈良県	奈良市	東紀寺町
6308306	ナラケン	ナラシ	キテラチョウ	奈良県	奈良市	紀寺町
6308307	ナラケン	ナラシ	ニシキテラチョウ	奈良県	奈良市	西紀寺町
6308311	ナラケン	ナラシ	ジユウリイノハタチョウ	奈良県	奈良市	十輪院畑田

6308296,ナラケン,ナラシ,ゴトウチョウ,
6308297,ナラケン,ナラシ,キタワハタチョウ,
6308301,ナラケン,ナラシ,タカバタケチョウ,
6308302,ナラケン,ナラシ,ヒヤクゴウジチョウ,
6308303,ナラケン,ナラシ,ミナミキテラチョウ,
6308304,ナラケン,ナラシ,ミナミカイノヅカチョウ,
6308305,ナラケン,ナラシ,ヒガシキテラチョウ,
6308306,ナラケン,ナラシ,キテラチョウ,
6308307,ナラケン,ナラシ,ニシキテラチョウ,

奈良県,奈良市,後藤町
奈良県,奈良市,北川端町
奈良県,奈良市,高畑町
奈良県,奈良市,白毫寺町
奈良県,奈良市,南紀寺町
奈良県,奈良市,南肘塚町
奈良県,奈良市,東紀寺町
奈良県,奈良市,紀寺町
奈良県,奈良市,西紀寺町

十輪院畑田
十輪院町
川之上町
川之上突抜
中辻町
築地之内田
井上町
納院町
薬師堂町

参考 <http://www.post.japanpost.jp/zipcode/dl/kogaki-zip.html>



国立大学法人奈良女子大学
Nara Women's University

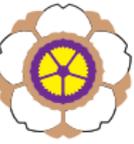
53

機能を考える

システムで実現する機能を考える

54

- 実現する機能について、優先度を決める
必須、**重要**、**後回し可** の3段階とする
- マスターデータの追加、変更、削除の機能の実現は、直接CSV形式のファイルに手を加えることで実現できるので後回しでもよい
- 在庫状況の確認は、遠隔地から携帯電話で確認できるのが理想だが、売主からの電話での問い合わせに対し、販売店でPOSシステムで確認した内容を電話で応えることができるものでもよい



マスターデータの登録、変更、削除機能

55

- 売主マスターデータ
 - 新たな売主のデータの登録 後回し可
 - 売主の登録データの変更 後回し可
 - 売主のデータの削除 後回し可
- 商品マスターデータ
 - 新たな商品のデータの登録 後回し可
 - 商品の登録データの変更 後回し可
 - 商品のデータの削除 後回し可

バーコードでの販売処理

56

□ 販売処理

- バーコードを読み取り、商品名と金額を表示する
- 複数購入する場合は、個数の入力に対応できる
- 購入した商品の合計金額を表示する

必須

重要

必須

□ 返品処理、返金処理

- 返品のリクエストがあった場合に返品と返金を行う

重要

□ 売値の変更処理

- 売主からの要望で売値を変更できる

必須

□ 商品の追加

- 売主が持ってきた商品を売り場に追加できる

必須

入荷処理(開店前)

57

- 商品の入荷に際して
 - ▣ 売値の確認、変更ができる
 - ▣ 在庫数の追加ができる

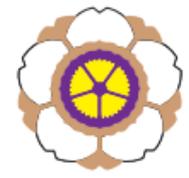
重要
必須

集計処理(閉店後)

58

- 閉店後のその日の売主ごとの集計処理
 - ▣ 商品ごとの販売数、売値、合計金額を表示する **必須**
- 売り上げ管理
 - ▣ その日の売り上げと利益の合計が表示される **必須**
- 統計処理
 - ▣ その月の販売数のデータを表示する **重要**
 - ▣ その月の販売数の推移をグラフで表す **後回し可**

- これ以外の機能を独自に追加しても構いません。
重要な機能が漏れているかも知れません。
- Python講習会の最終日までに、少なくとも**必須**の機能はシステムとして実現してください。
- **重要**の機能については、時間があれば、作成してください。



ファイルを読み込むプログラムの作成

ファイルの読み込み

61

- ファイルを読み込む前に、『open』を実施します。
- 1行ずつ読み込み、変数に代入するためには、『for』で繰り返しを行う必要があります。
- 例えば、data01.csvというファイルが下のような場合

1001, orange, 80

1002, apple, 120

1003, banana, 100

```
# ファイルのオープン
bcode_file = open( 'data01.txt' , 'r' )

# 変数の初期化
bcode = []
i = 0

# ファイルの内容を読み込み、二次元配列の変数に代入する
for line in bcode_file:
    bcode.append( line.strip().split(',') )
    print( bcode[i][1] )
    i += 1
```

- 実行すると、果物の名前が表示されます。

サンプルプログラム

63

```
# ファイルのオープン
bcode_file = open( 'data01.txt' , 'r' )

# 変数の初期化
bcode = []
i = 0
total = 0

# ファイルの内容を読み込み、二次元配列の変数に代入する
for line in bcode_file:
    bcode.append( line.strip().split(',') )
    print( bcode[i][1] )
    i += 1

# 無限ループ
```

以下は次のページに続く

サンプルプログラム(続き)

64

```
# 無限ループ
while True:
    # バーコードリーダーからの入力
    code = input()
    print( code )
    # もしバーコードの値が終了コードなら無限ループを抜ける
    if code == '999':
        break
    else:
        # data を初期化
        data = [ 9999 , 'not found', 0 ]
        # bcodeから同じバーコードを検索する
        for line in bcode:
            # 見つかったらdataにデータをセットする
            if code in line:
                data = line
        print( 'name ' , data[1] , 'price ' , data[2] , '円' )
        total += int ( data[2] )
print ( '合計金額' , total , '円' )
```

時刻の取得

65

```
import datetime
today = datetime.datetime.now()
print( today )
print( today.weekday() )
print( today.strftime( '%Y/%m/%d' ) )
```

□ 実行すると

```
2016-12-22 11:39:27.742481
3
2016/12/22
```

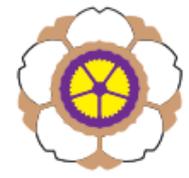
月曜	0
火曜	1
水曜	2
木曜	3
金曜	4
土曜	5
日曜	6

と表示される

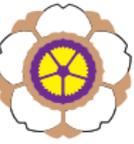
```
# ファイルのオープン
bcode_file = open( 'data01.txt' , 'r' )

# 変数の初期化
bcode = []
i = 0

# ファイルの内容を読み込み、二次元配列の変数に代入する
for line in bcode_file:
    bcode.append( line.strip().split(',') )
    print( bcode[i][1] )
    i += 1
```



本日はここまで



プログラミング講座のスケジュール

68

□ 2日目

- 午前 商品の販売を行うPOSレジの作成
- 午後 商品の販売状況を確認するプログラムの作成
プログラムの発表会(くじ引き)