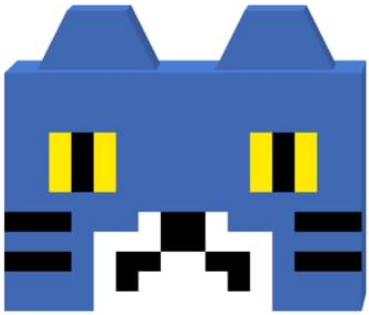
マンガでわかる Python

基本の基本を知ろう
データ型
複数の値のデータ型
条件分岐をしよう
繰り返し処理をしよう
関数を作ろう
エラーとエラー制御
ファイルの読み書きをしよう

目次

第1章

プロローグ



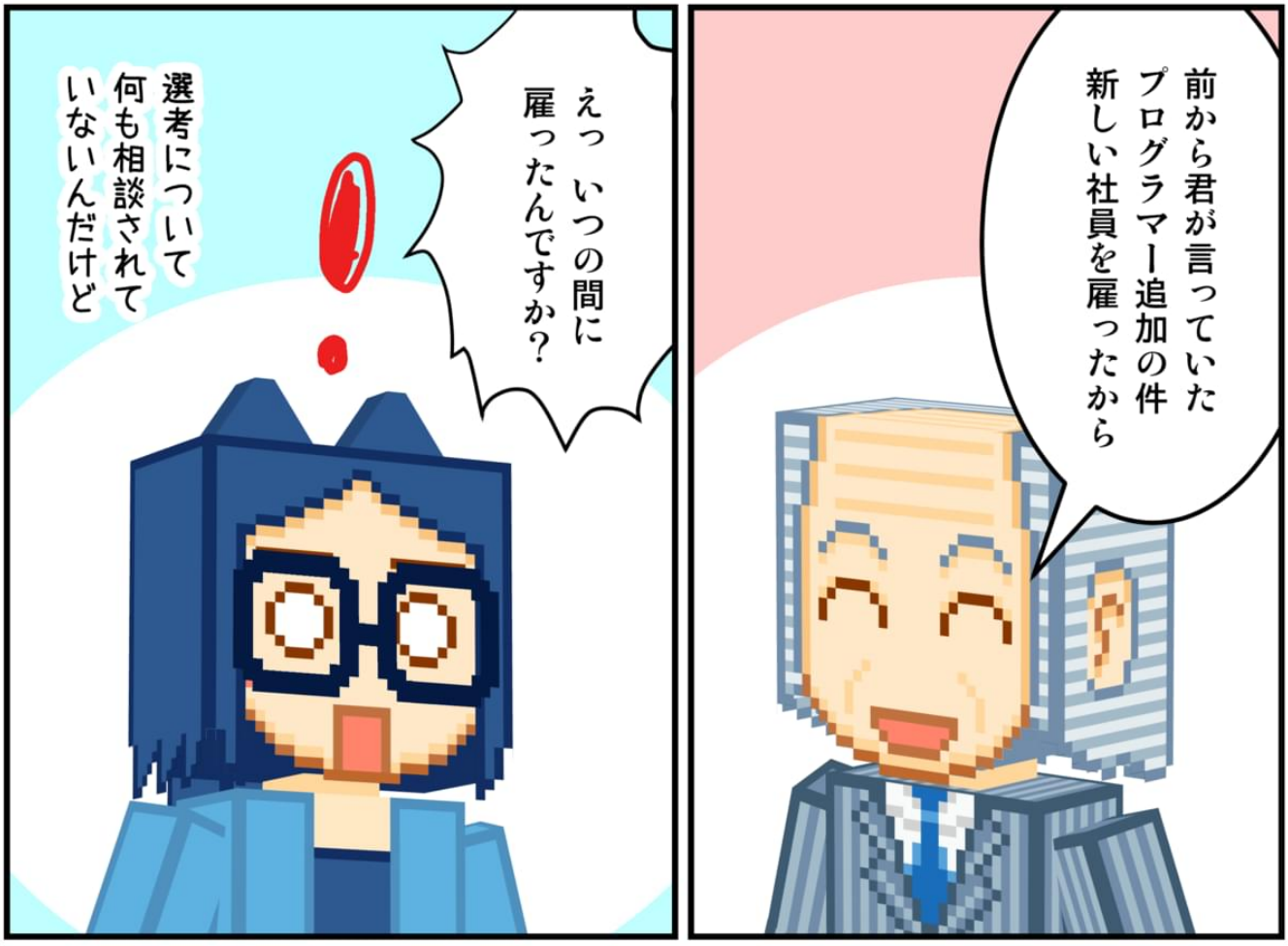
1-1「新人教育」

忠商企業
株式会社


カタカタ


猫野くん
今話しかけても
大丈夫かね？
忠商企業社長
長久務
ながく つとむ


何ですか社長？
開発室長
（部下なし）
猫野美実
ねこの みみ


前から君が言っていた
プログラマー追加の件
新しい社員を雇ったから
えっ いつの間に
雇ったんですか？
選考について
何も相談されて
いないんだけど


この子だよ
新見入子です！
にいみ
いりこ
よろしく
お願いします！

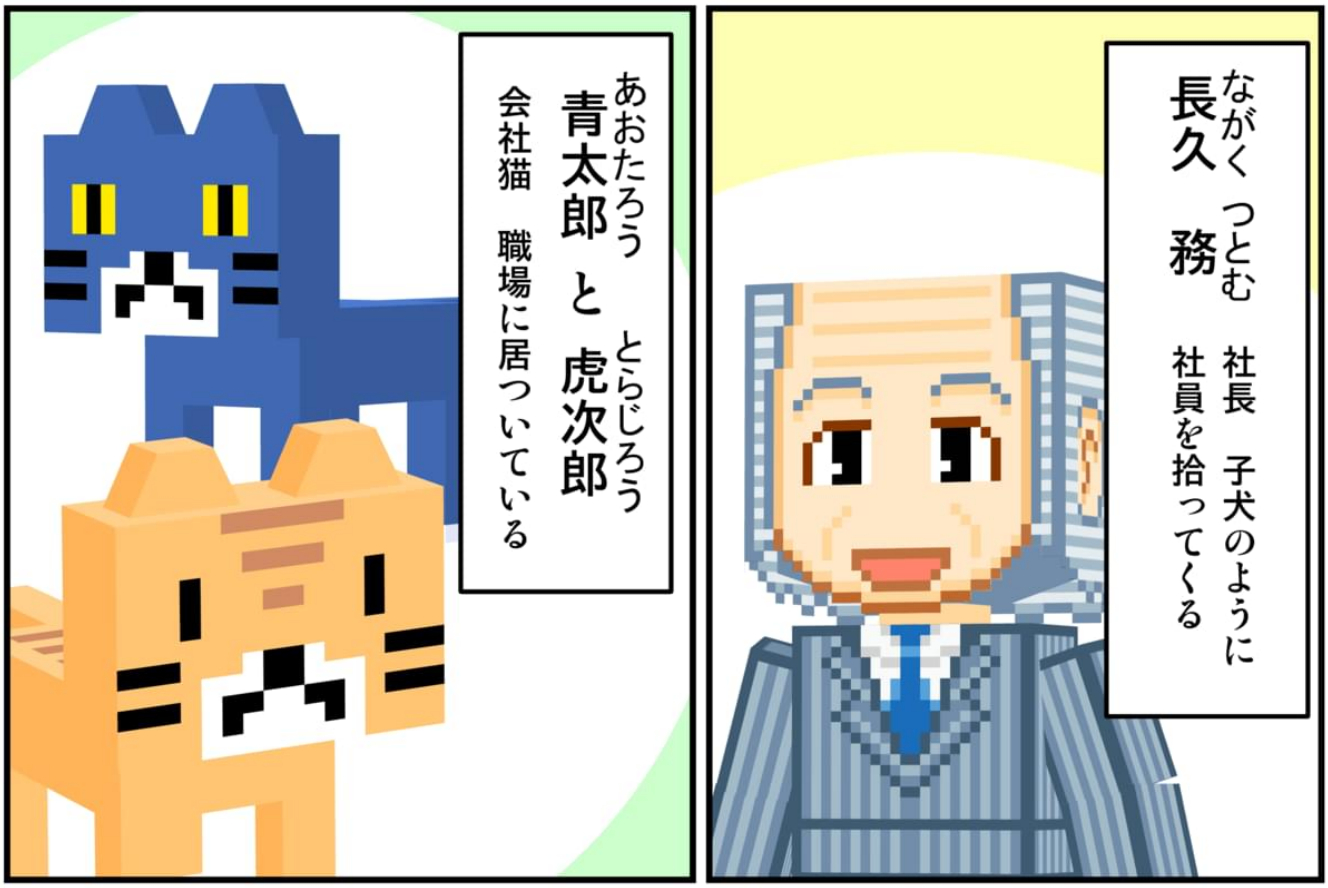

新見さんの
開発歴は？
？
新見くんは
新卒後、就職に失敗して
バイトで生活していた
頑張り屋さんだよ
にこにこ
嫌な予感がする

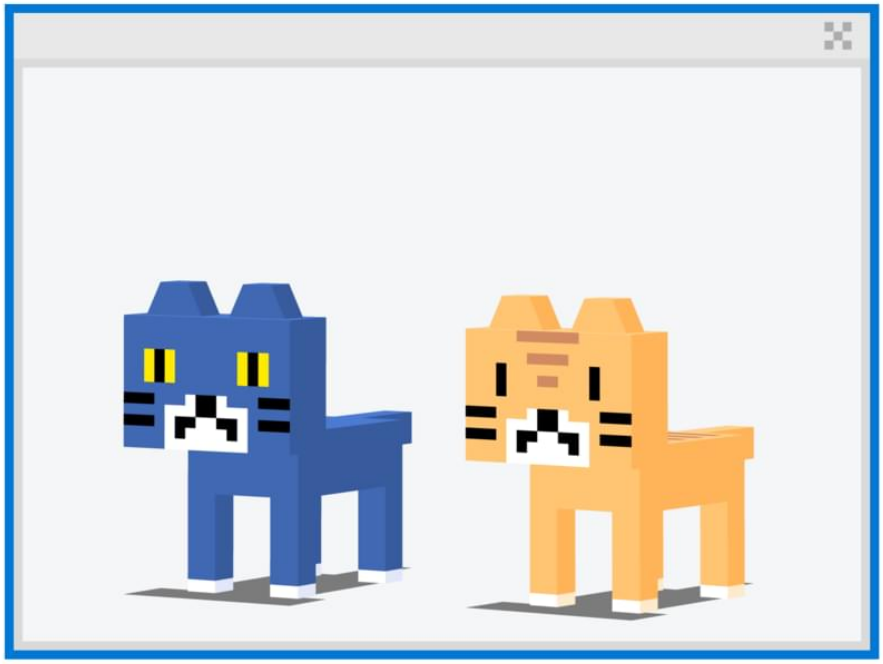

たまたま
そば屋で話を聞いてねえ
引き抜いてきたんだ
この社長
またやらかし
やがった
先輩！
よろしく
お願いします！
私は何からやれば
いいんですか！
あ、うん、
そうだね…


というわけで
プログラミングの
教育は頼んだよ
猫野くん
これで戦力は
2倍になるね
はっ
ははっ…
ところで先輩
プログラミングって
何ですか？
うん、じっくり
勉強していこうか
おわり

忠商企業　登場人物

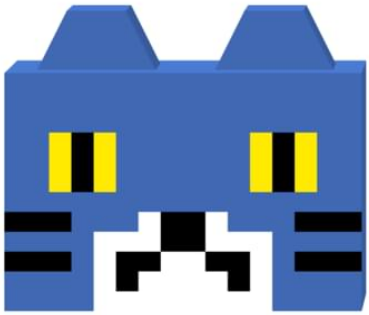

ねこの みみ
猫野 美実
開発室長　ワンオペに
苦しんでいたプログラマー
にいみ いりこ
新見 入子
新入社員　頑張り屋さん
プログラミング未経験


ながく つとむ
長久 務
社長　子犬のように
社員を拾ってくる
あおたろう と とらじろう
青太郎 と 虎次郎
会社猫　職場に居ついている

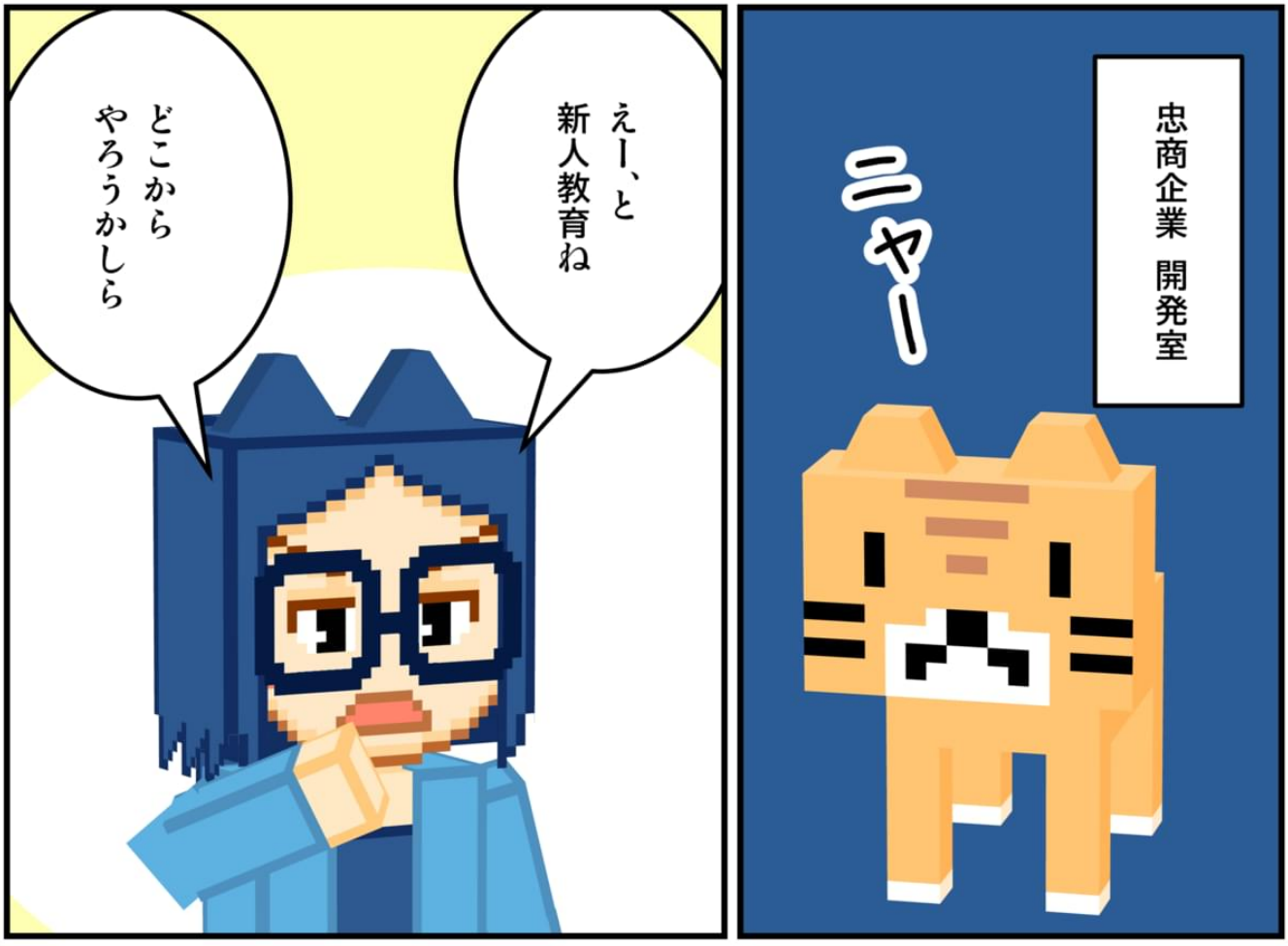


第2章

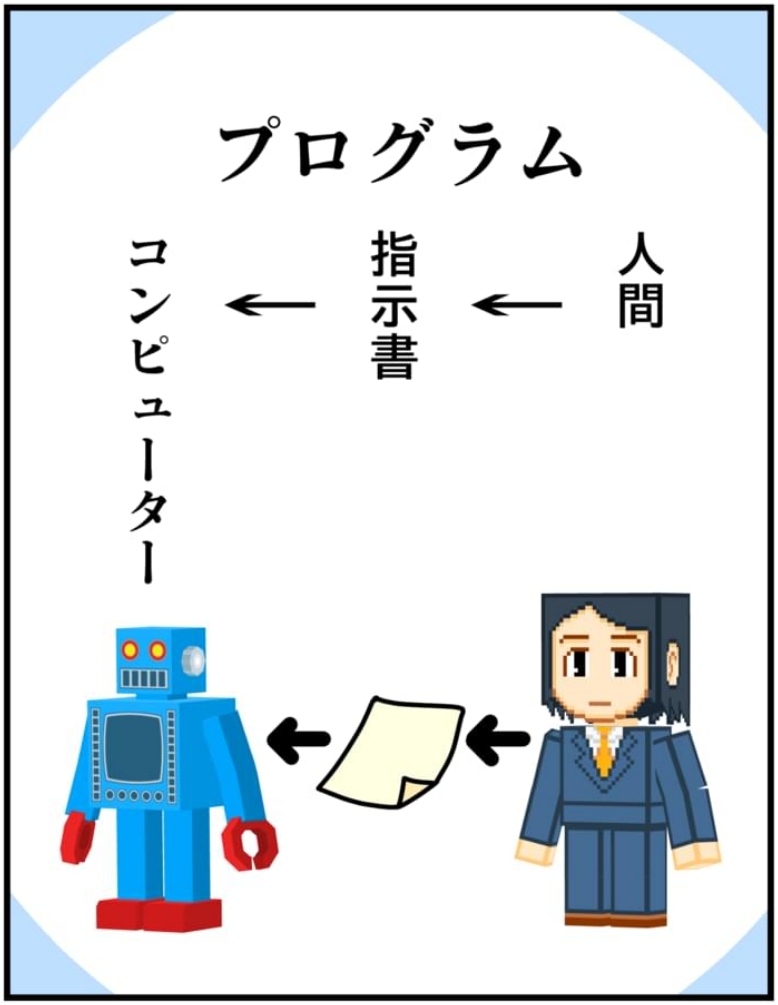
まずは下準備



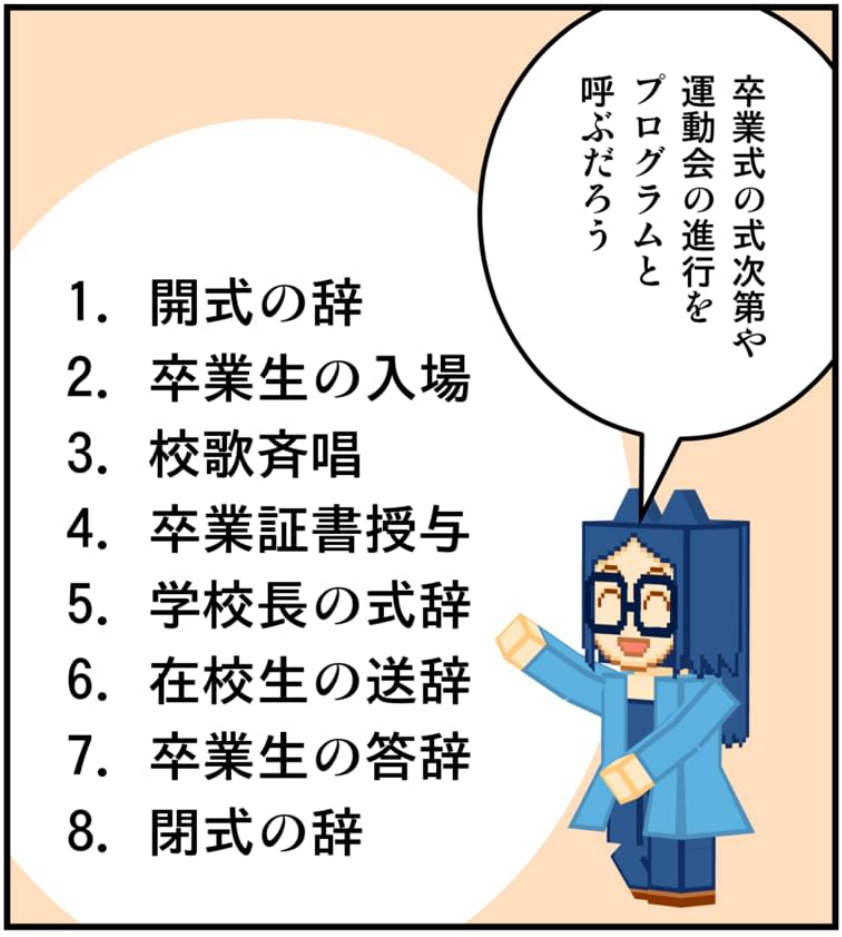
2-1「プログラムって何？」


忠商企業　開発室
ニャー
えー、と
新人教育ね
どこから
やろうかしら


はい、先輩！
そもそも
プログラムって
何ですか？
そうだよなあ
そこからだよなあ


プログラム
人間
←
指示書
←
コンピューター

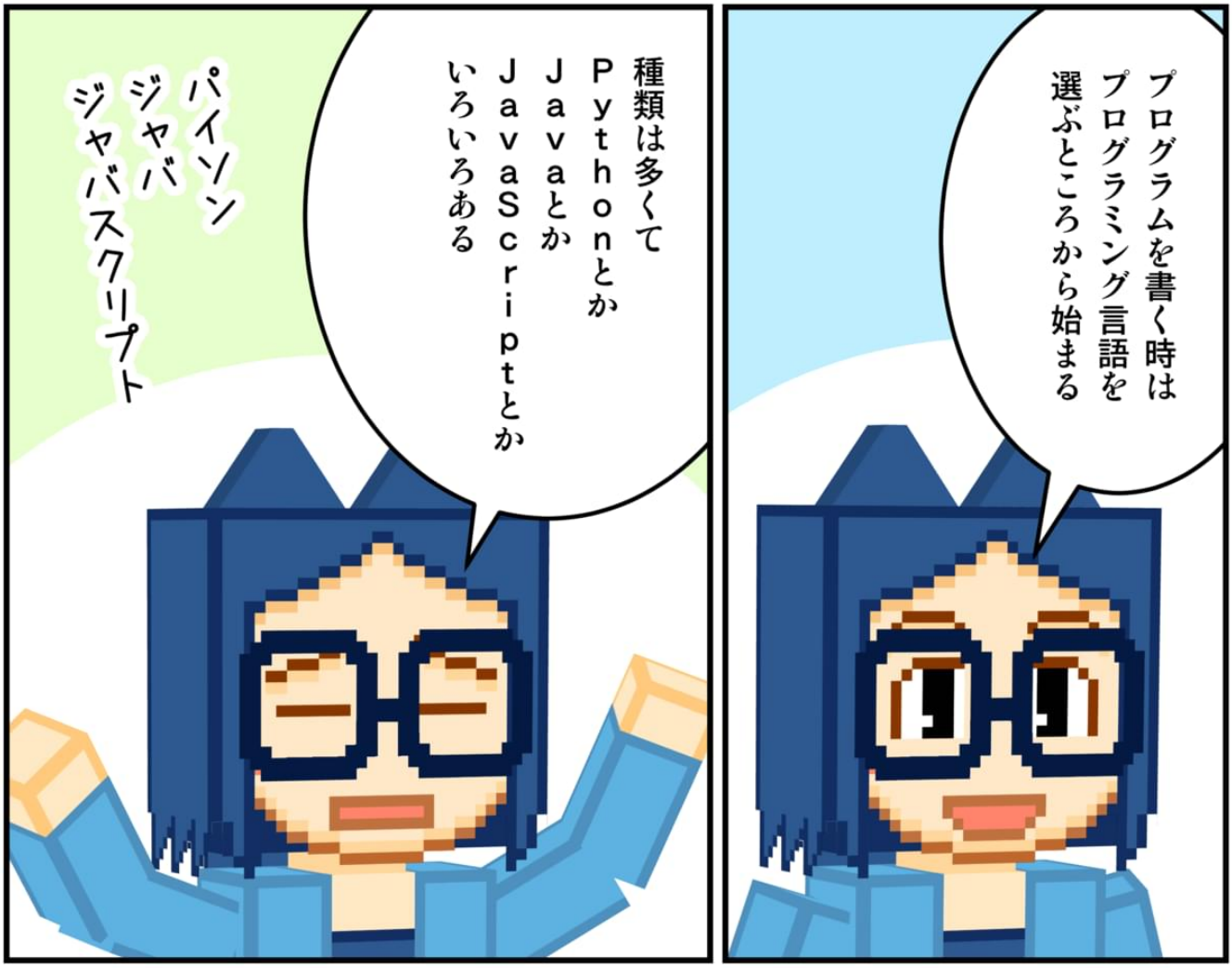

プログラムは
コンピューターへの
指示書だ


卒業式の式次第や
運動会の進行を
プログラムと
呼ぶだろう
1. 開式の辞
2. 卒業生の入場
3. 校歌斉唱
4. 卒業証書授与
5. 学校長の式辞
6. 在校生の送辞
7. 卒業生の答辞
8. 閉式の辞

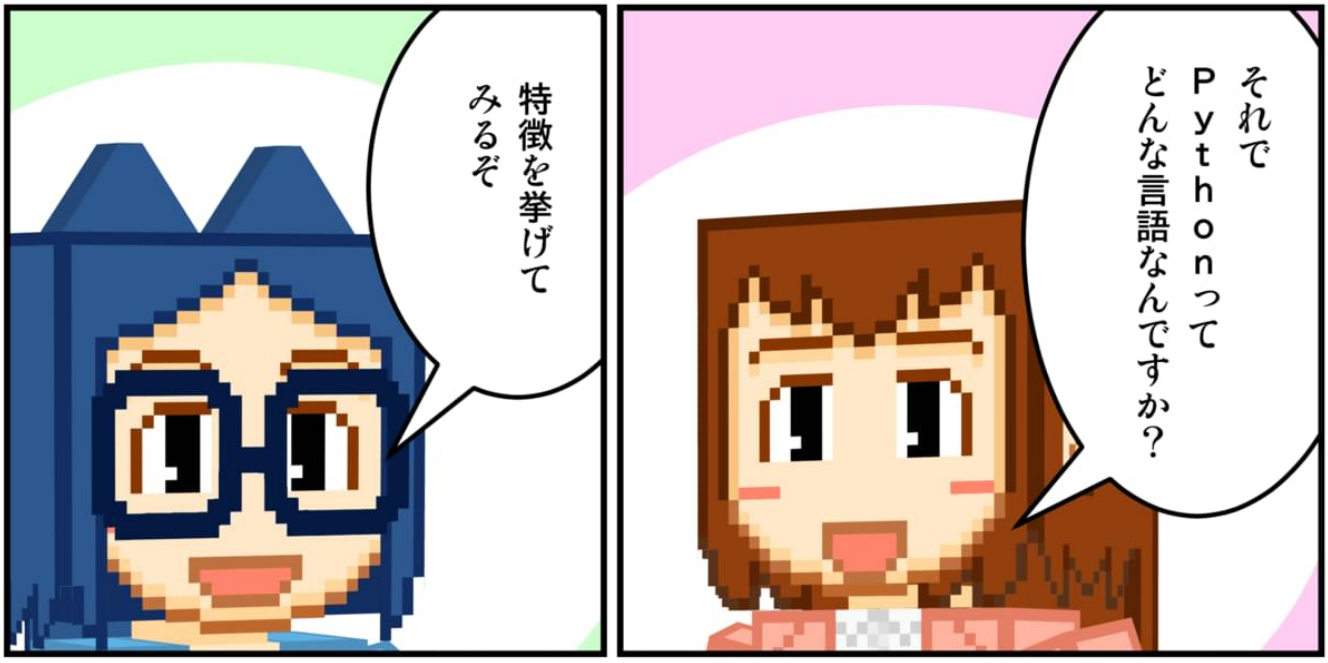

ああいった
何を順番におこなうのかを
書いた文書のことを
プログラムと言うんだ


あと
プログラムを書くことを
プログラミング
プログラムを書く人を
プログラマーと呼ぶ
そして
プログラムを
書く言葉を
プログラミング
言語と呼ぶ

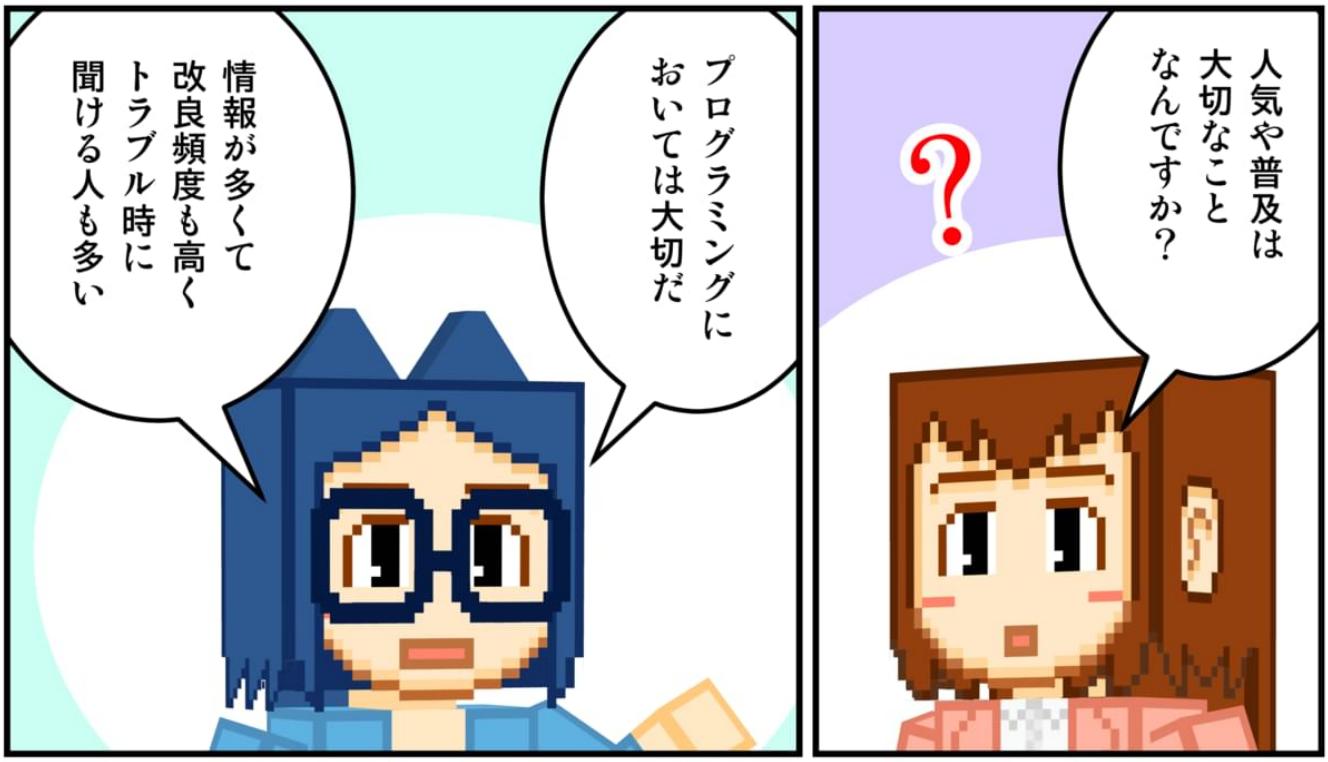

中国語や
フランス語みたいに
プログラミング国の
言葉ですか？
いや違う
プログラム用の
特別な記法のことだ
文法が決まっていて
さまざまな命令が
用意されている

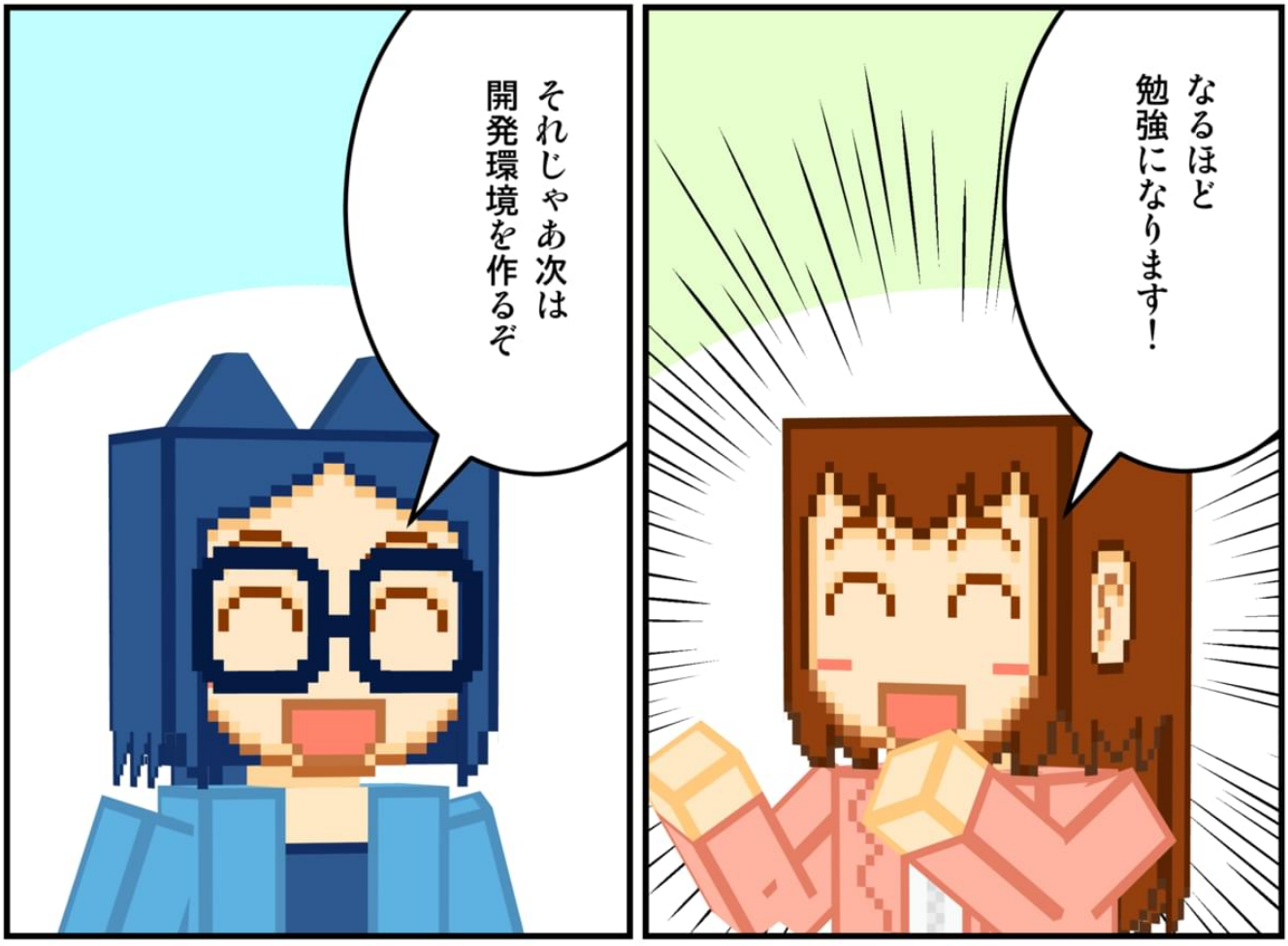

プログラムを書く時は
プログラミング言語を
選ぶところから始まる
種類は多くて
Ｐｙｔｈｏｎとか
Ｊａｖａとか
ＪａｖａＳｃｒｉｐｔとか
いろいろある
パイソン
ジャバ
ジャバスクリプト


なるべく
簡単そうなので
お願いします
そこは考えている
Ｐｙｔｈｏｎを
教えようと思う

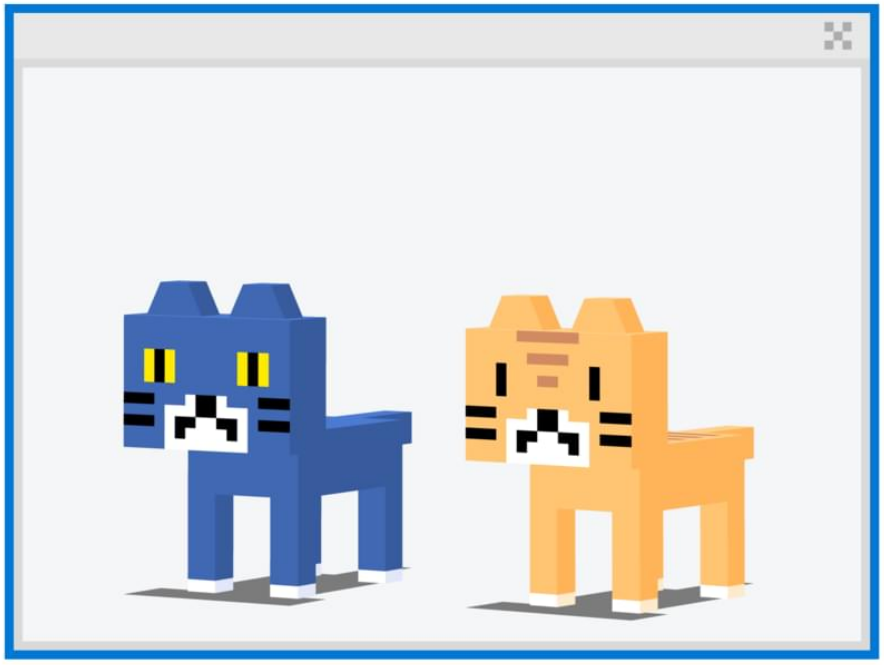


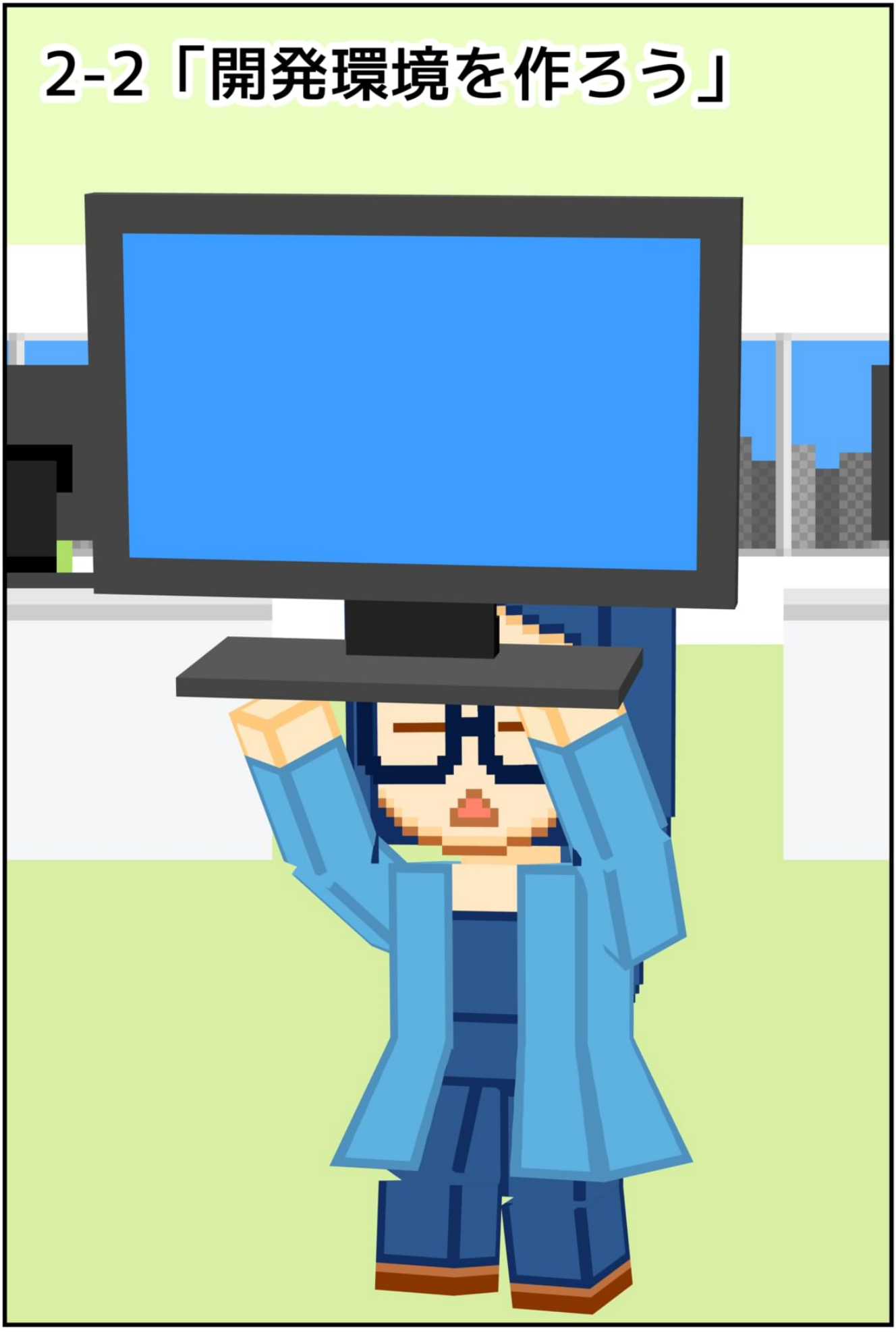
・平易で分かりやすい
・人気がある
・普及している




なるほど
勉強になります！
それじゃあ次は
開発環境を作るぞ


というわけで
少し休憩だ
ニャー



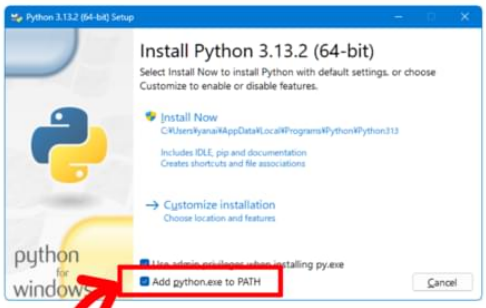

2-2「開発環境を作ろう」


それじゃあ
開発環境を作るぞ
パソコンを
用意しろ
開発環境を作るって
何をするんですか？


Pythonの
プログラムを
実行できるソフトを
インストールする
あと
プログラムを書く
エディターを
インストールする
分かりました

Python実行用のソフト
https://www.python.org/
downloads/

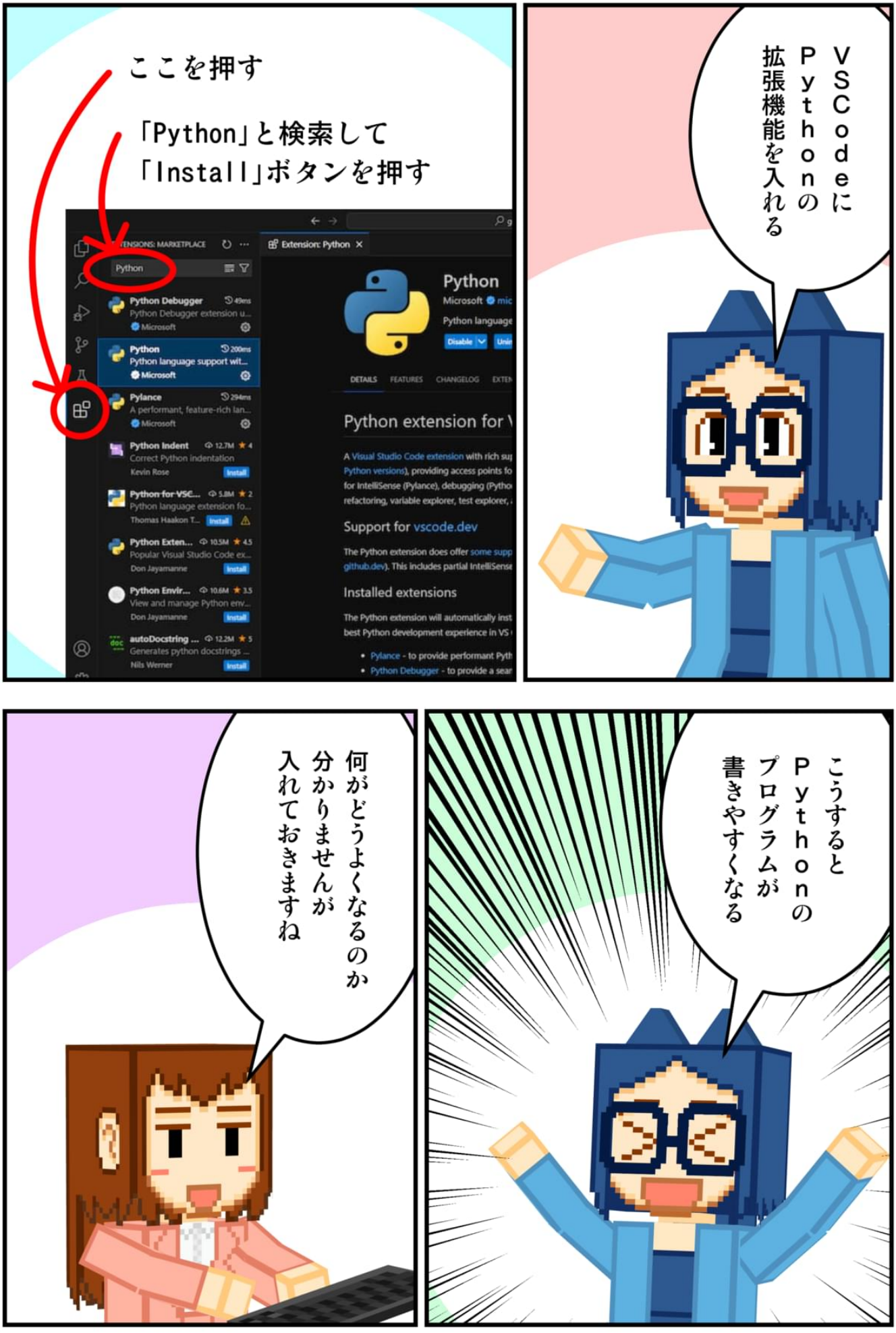
インストールダイアログ

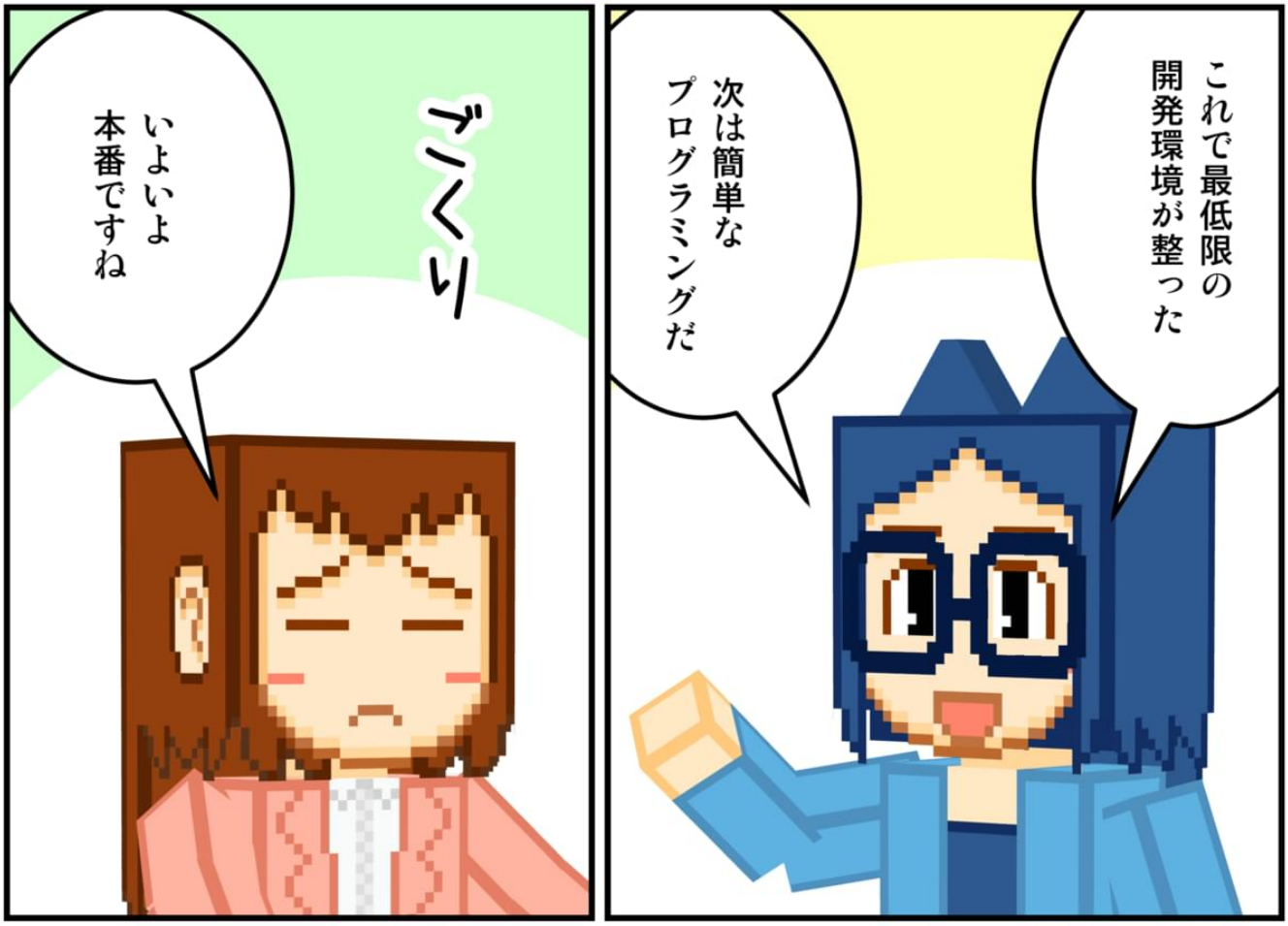


Windowsなら
ここにチェックを入れる

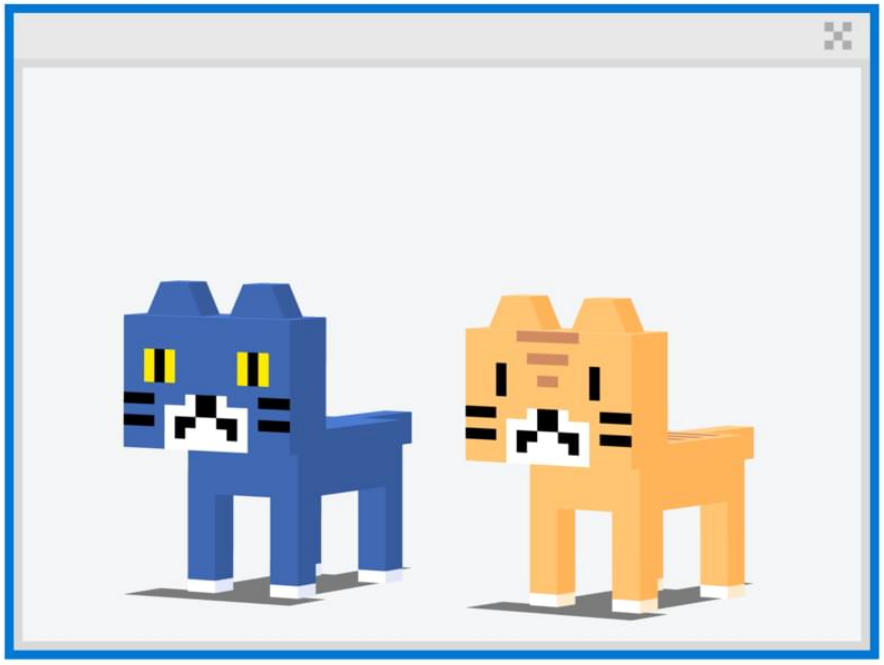
プログラミング用のエディター
Visual Studio Code
https://code.visualstudio
.com/Download

VSCodeと略すことが多い

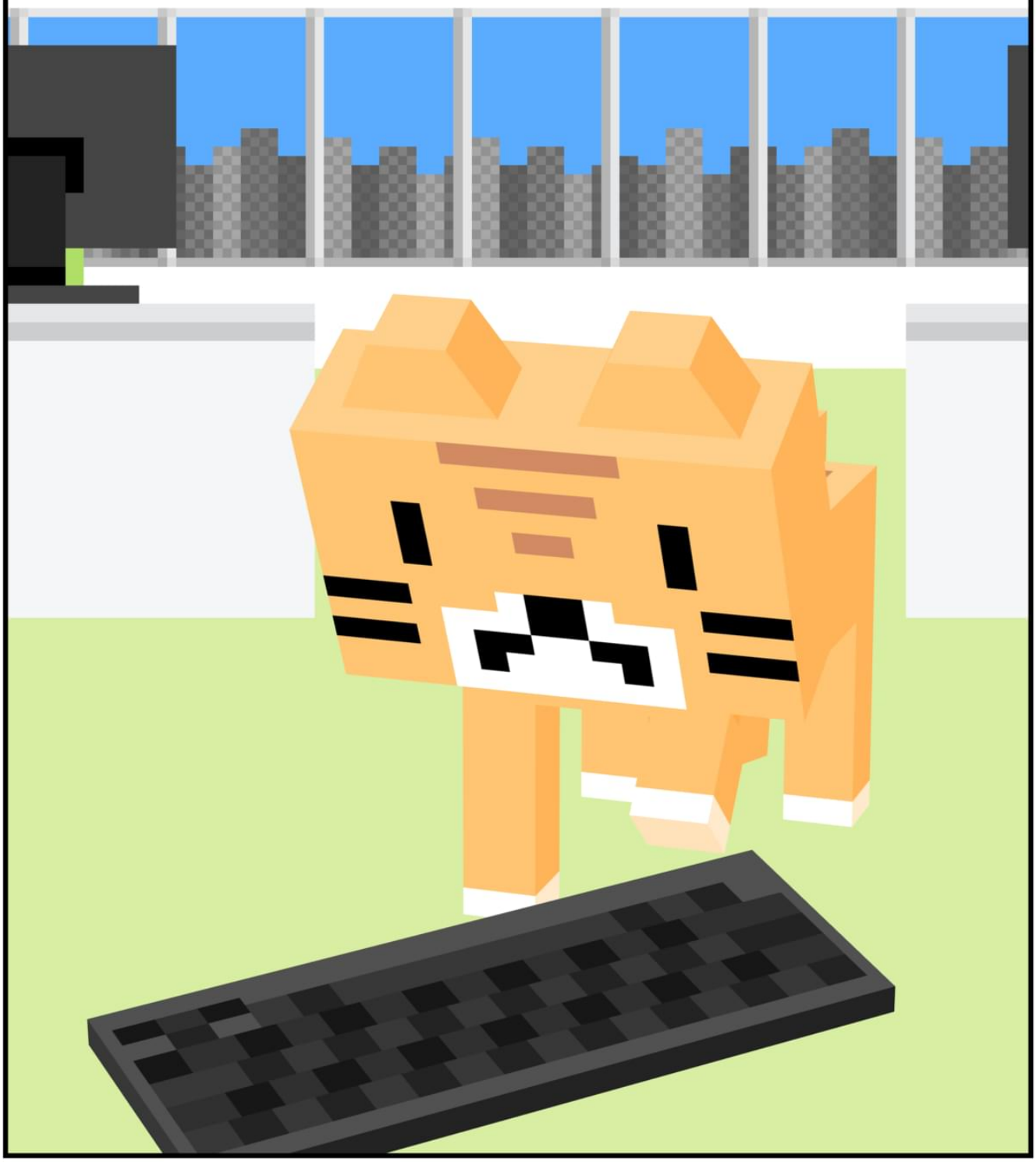

ここを押す
「Python」と検索して
「Install」ボタンを押す
Python
Python Debugger
Python language support wit...
Microsoft
Pylance
Python Indent
Correct Python indentation
Kevin Rose
Python for VSC...
Python language extension fo...
Thomas Haakon T...
Python Exten...
Popular Visual Studio Code ex...
Don Jayamanne
Python Envir...
View and manage Python env...
Don Jayamanne
autoDocstring ...
Generates python docstrings ...
Nils Werner
Extension: Python
Python
Microsoft
Python language
Disable
DETAILS
FEATURES
CHANGELOG
Python extension for
Support for vscode.dev
Installed extensions
VSCodeにPythonの拡張機能を入れる
何がどうよくなるのか分かりませんが入れておきますね
こうするとPythonのプログラムが書きやすくなる


これで最低限の
開発環境が整った
次は簡単な
プログラミングだ
ごくり
いよいよ
本番ですね


あまり
緊張しなくていい
猫でもなでて
リラックスしてくれ
なで〜



2-3「簡単なプログラムの実行」



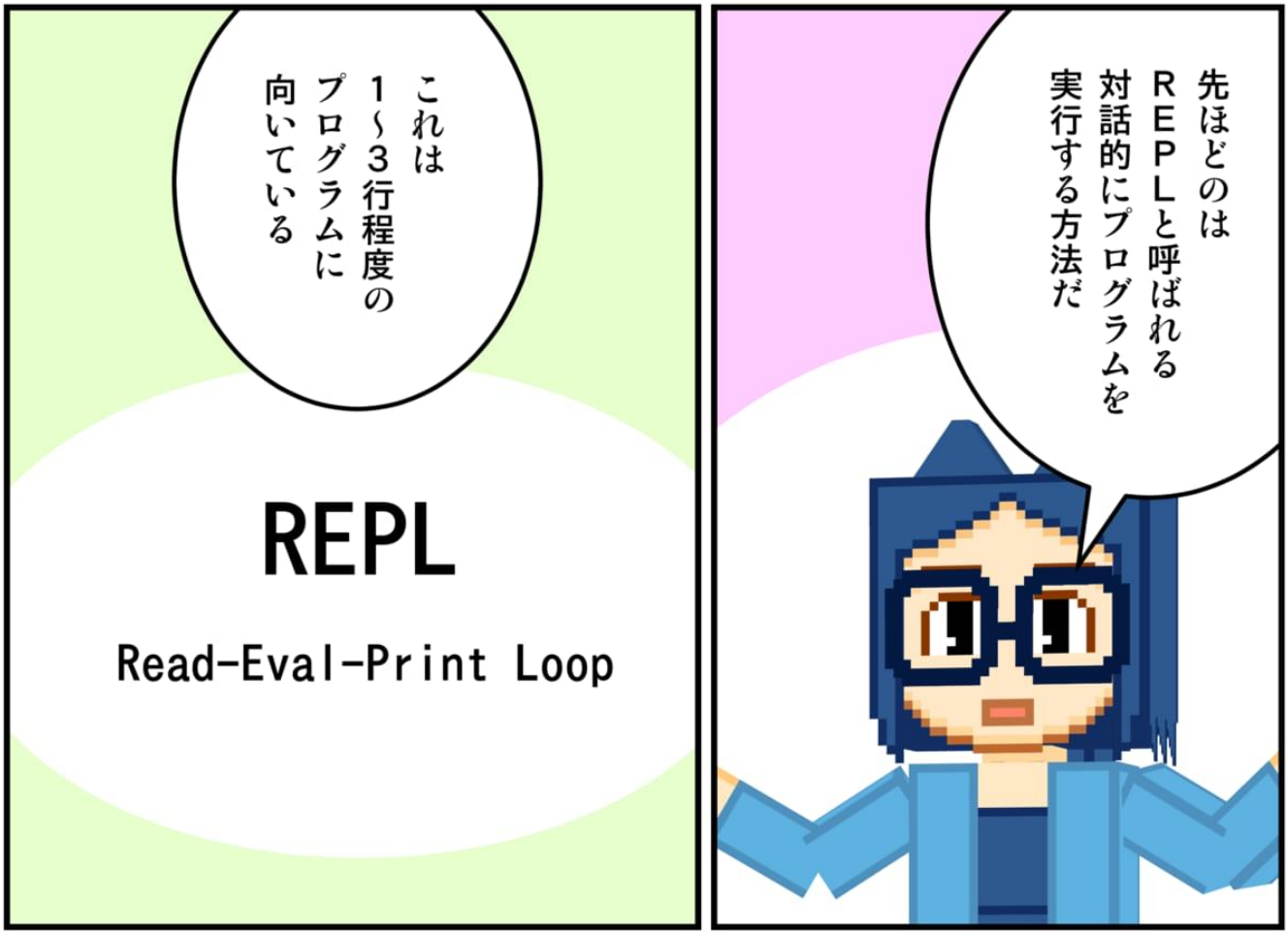

それじゃあ簡単な
プログラムを書いて
実行してみるぞ
いよいよですね
VSCodeで
ターミナルを開く
[Ctrl+@]あるいは
メニューの「View＞Terminal」で
ターミナルを開く
CtrlはMacでは⌘(Cmd)
そして
Pythonを
実行する
「python」と入力して
[Enter]キーを押すと
行の左端が「>>>」になる
pythonコマンドは
Macだとpython3に読み換え
PROBLEMS
OUTPUT
TERMINAL
PORTS
DEBUG CONSOLE
PS C:\Users\yana> python
Python 3.13.2 (t
f8bb39, Feb 4 2025, 15:23:48) [MSC v.
lp", "copyright", "credits" or "license" for more informati
>>>

「1+2」と半角文字で入力して
[Enter]キーを押す

```
>>> 1+2
3
```

計算結果の「3」が表示される

[Ctrl+Z]で終了できる

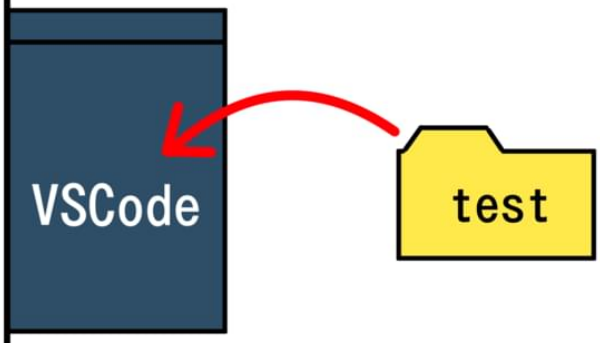

先ほどのは
REPLと呼ばれる
対話的にプログラムを
実行する方法だ
これは
1〜3行程度の
プログラムに
向いている
REPL
Read-Eval-Print Loop


それより長い
プログラムは
ファイルに書いて
実行する
この方法も
説明するぞ
分かりました

それでは
手順を説明するぞ

まず、プログラム用の
フォルダーを作り
ＶＳＣｏｄｅで開く

「test」フォルダーを作る
VSCodeのウィンドウに
ドラッグ＆ドロップする



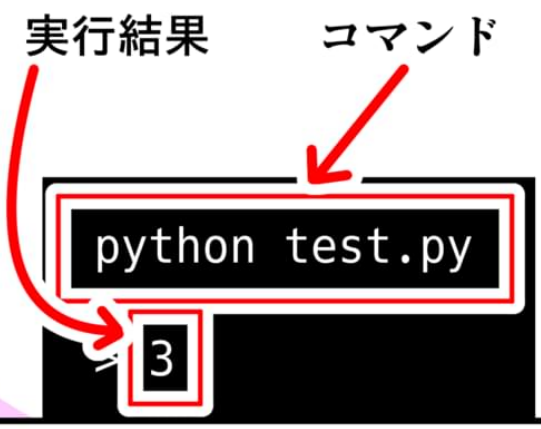
フォルダー内に
ファイルを作り
ＶＳＣｏｄｅで開いて
プログラムを書いて
保存する

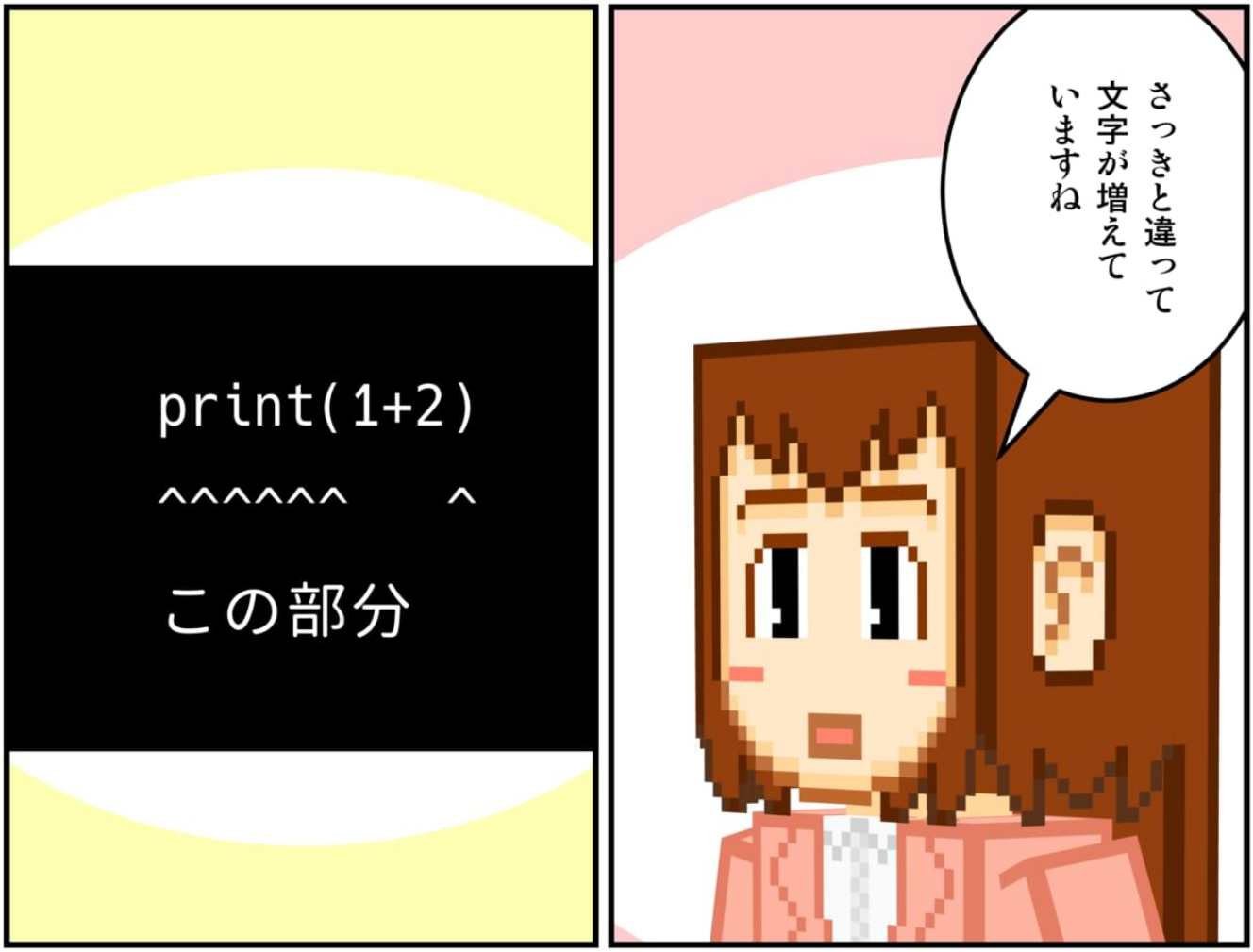
拡張子が.pyのファイル
ここでは「test.py」にする

「test.py」の中身

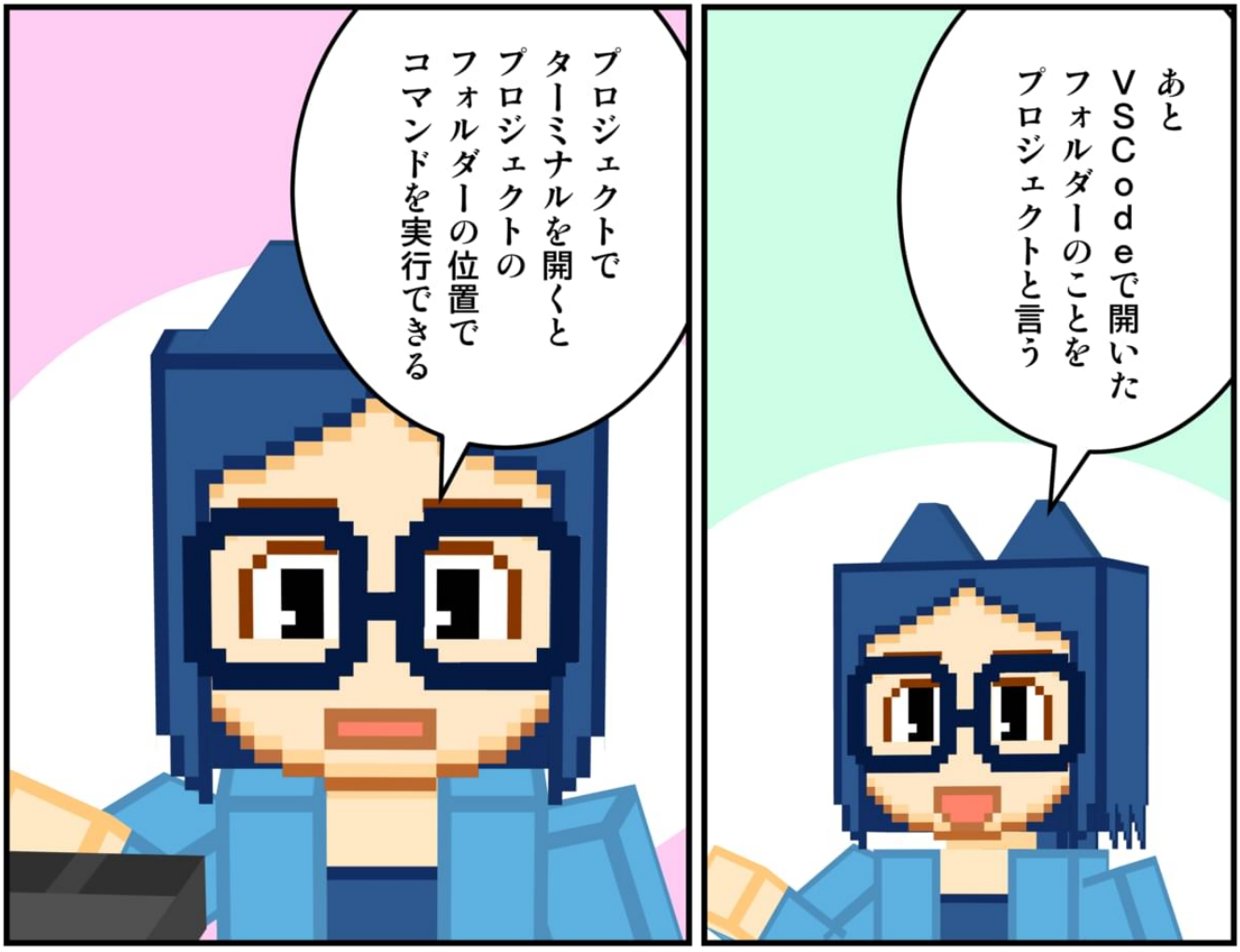
```
print(1+2)
```

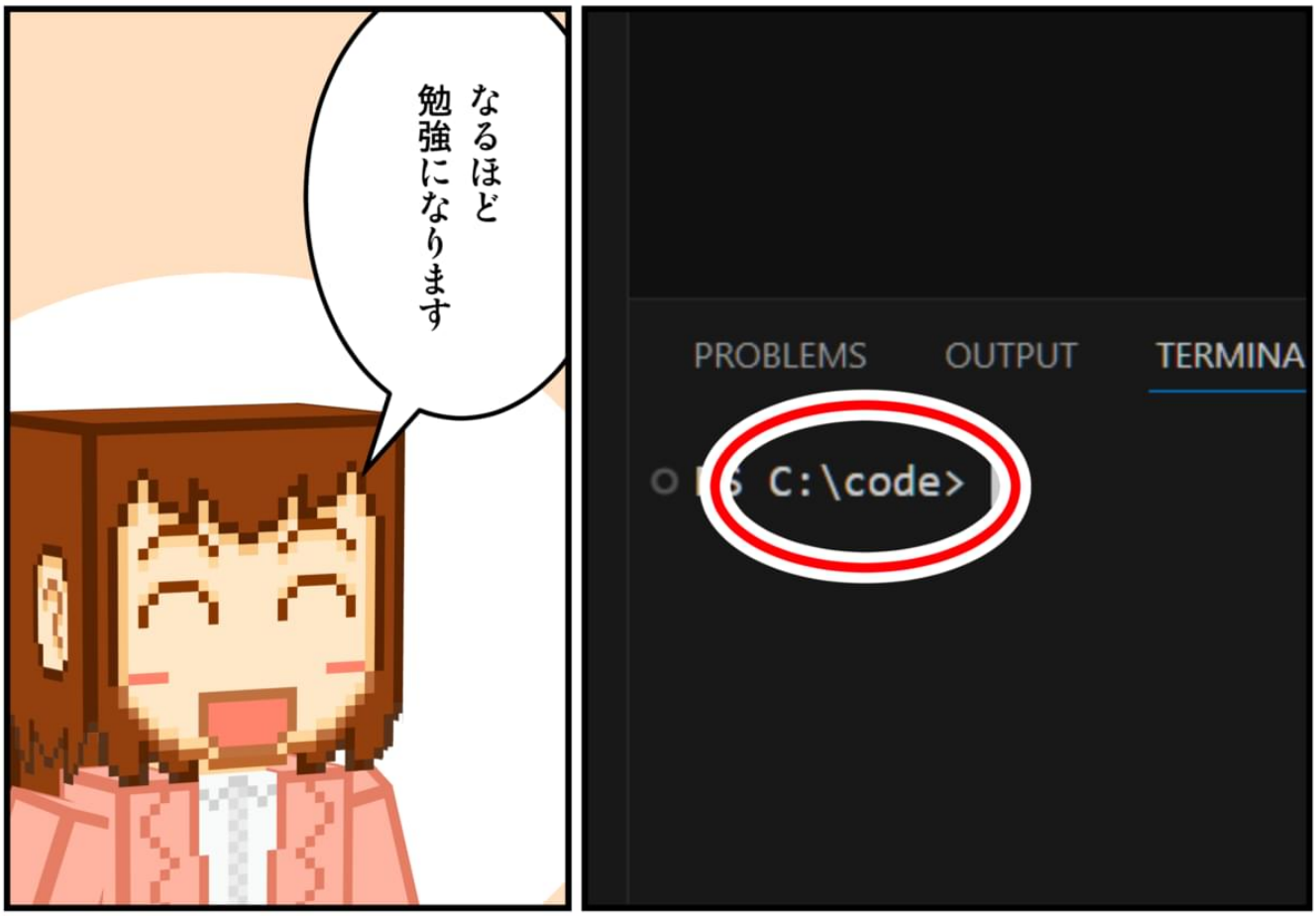
ターミナルを開き
コマンドを書いて
Ｅｎｔｅｒキーを
押して実行する




さっきと違って
文字が増えて
いますね
print(1+2)
^^^^^^　　^
この部分

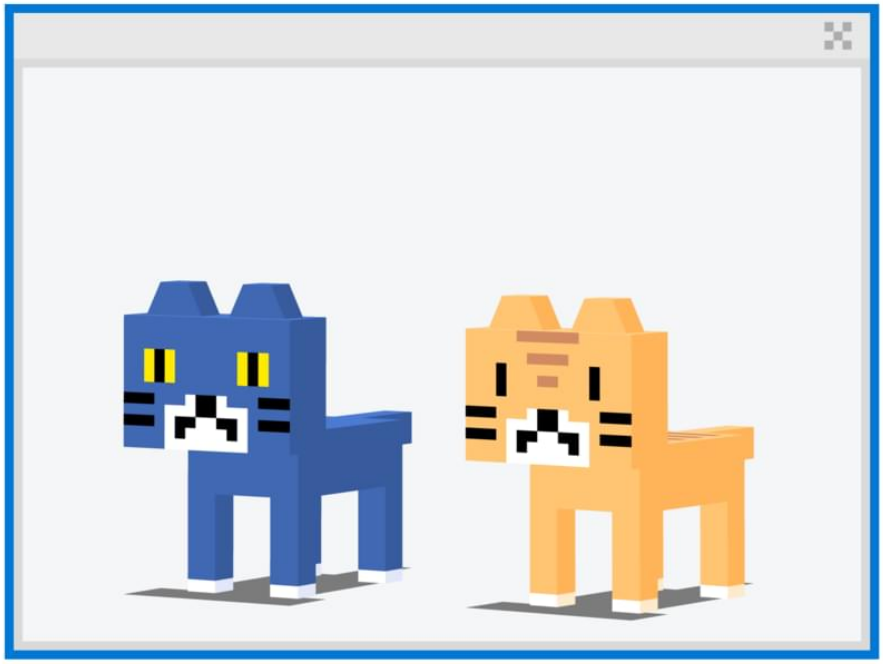

REPLでは
結果が直接
表示された
ファイルでは
画面に表示するには
print()と
書かないといけないんだ


あと
VSCodeで開いた
フォルダーのことを
プロジェクトと言う
プロジェクトで
ターミナルを開くと
プロジェクトの
フォルダーの位置で
コマンドを実行できる


なるほど
勉強になります
PROBLEMS
OUTPUT
TERMINA
C:\code>


とりあえず
ここまでで
プログラムを書いて
実行できる環境が
整った
やったー

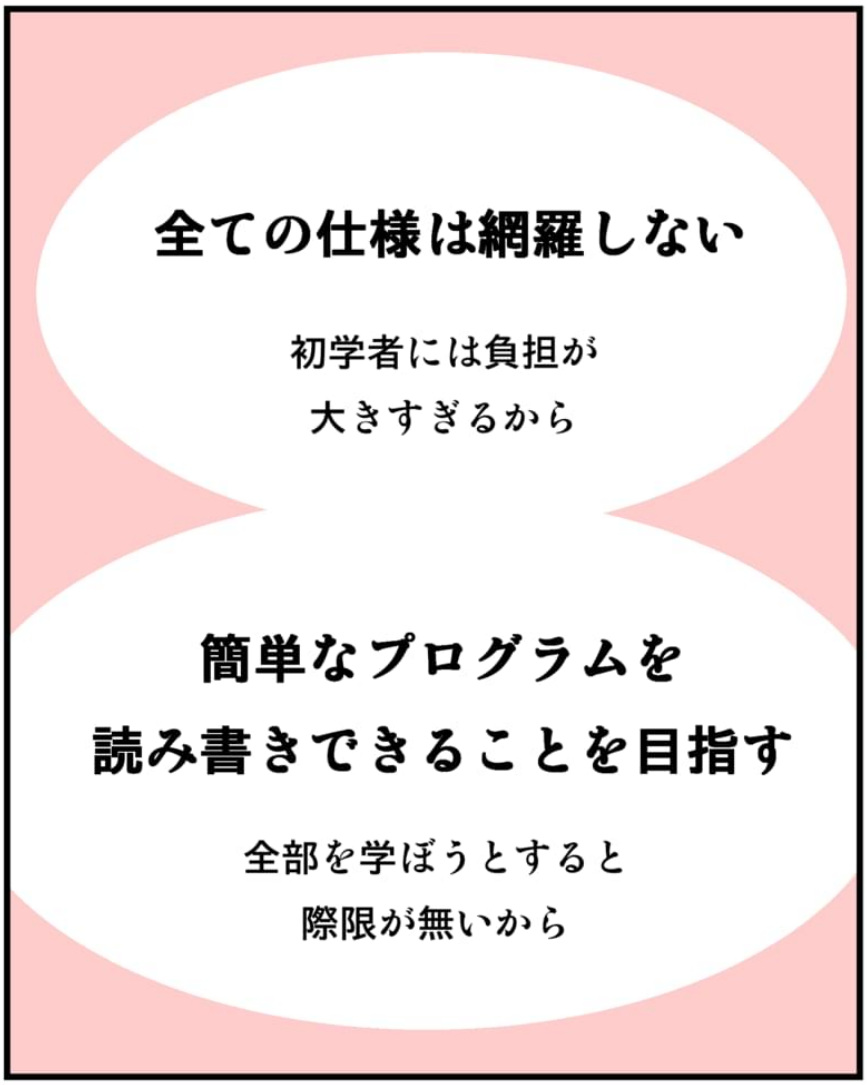

これで本格的な
Ｐｙｔｈｏｎの勉強を
始められるぞ
はい、先輩
よろしく
お願いします！

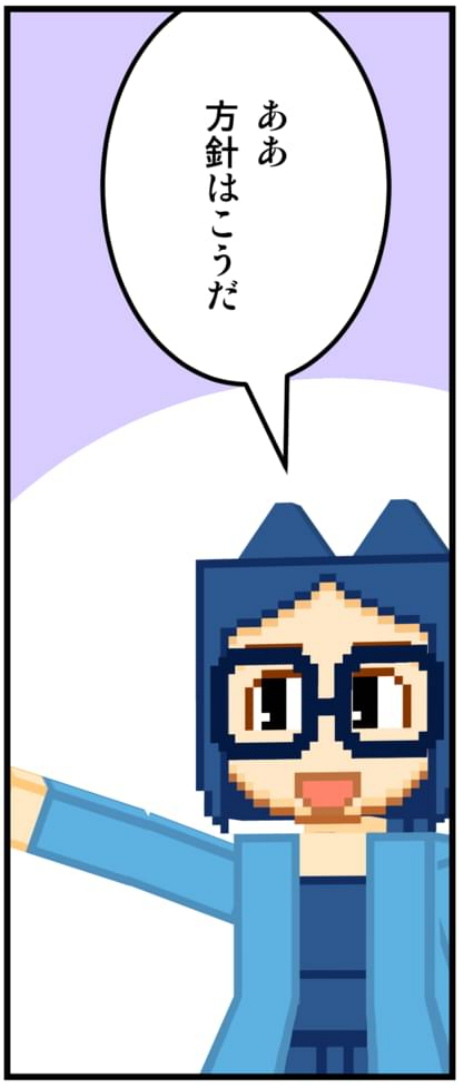


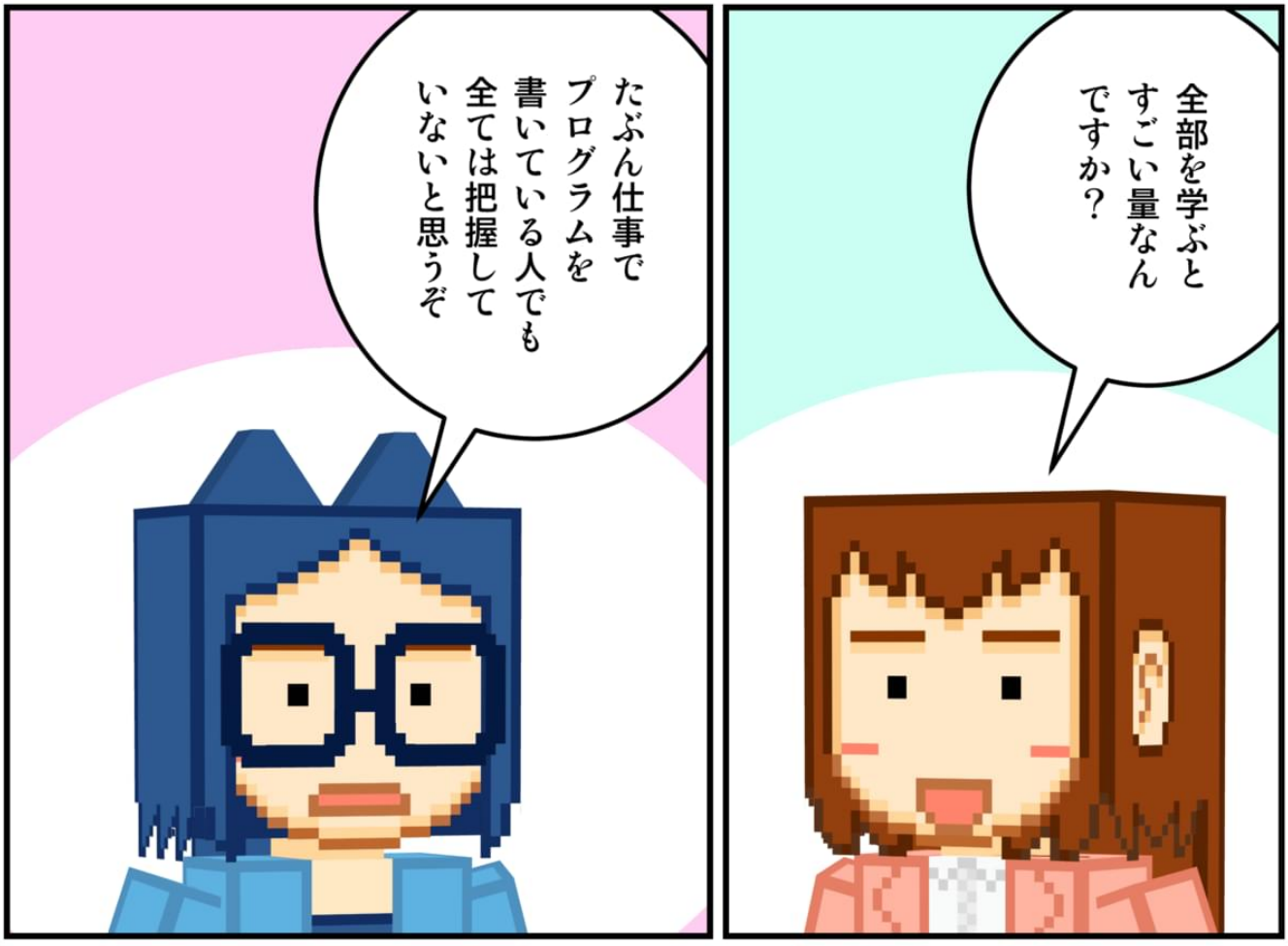
2-4「今後の方針」


準備はできたから
今後の勉強の方針を
示すぞ

まずは大枠を
つかむんですね

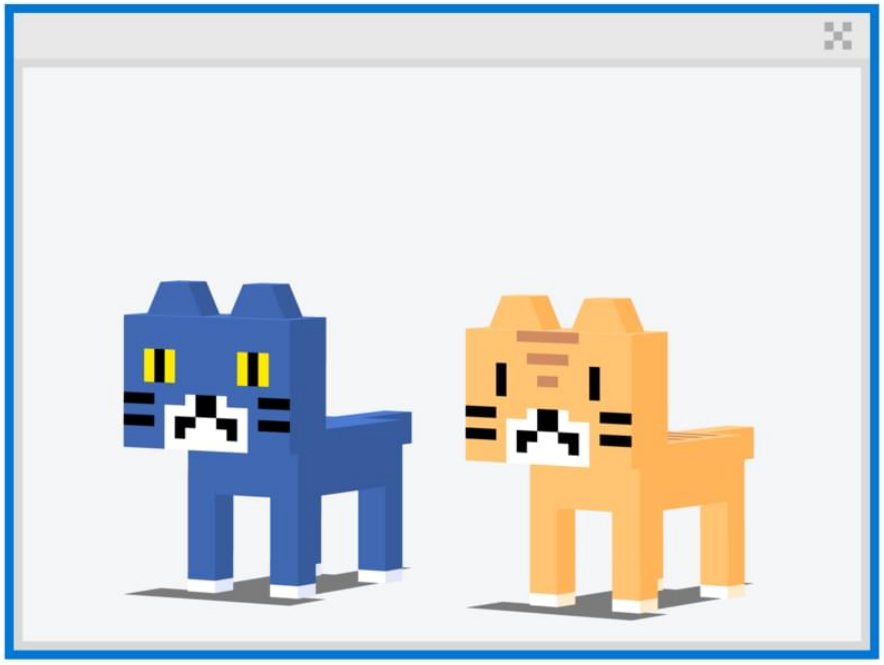

全ての仕様は網羅しない
初学者には負担が
大きすぎるから
簡単なプログラムを
読み書きできることを目指す
全部を学ぼうとすると
際限が無いから


ああ
方針はこうだ


全部を学ぶとすごい量なんですか？
たぶん仕事でプログラムを書いている人でも全ては把握していないと思うぞ

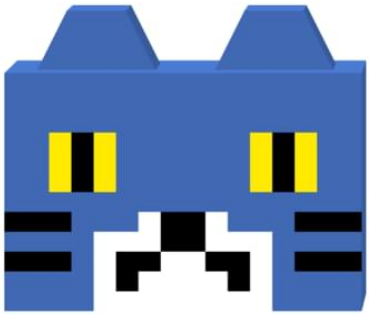

じゃあプログラマーの人はどうしているんですか？
その都度調べる
ただ、調べるにも基礎的な知識が必要だ


というわけで
大変だけど
付いてきてくれ
はい
戦力になるように
頑張ります！
ああ
戦力になるのは
いつぐらいだろう
まじで
大丈夫かなあ

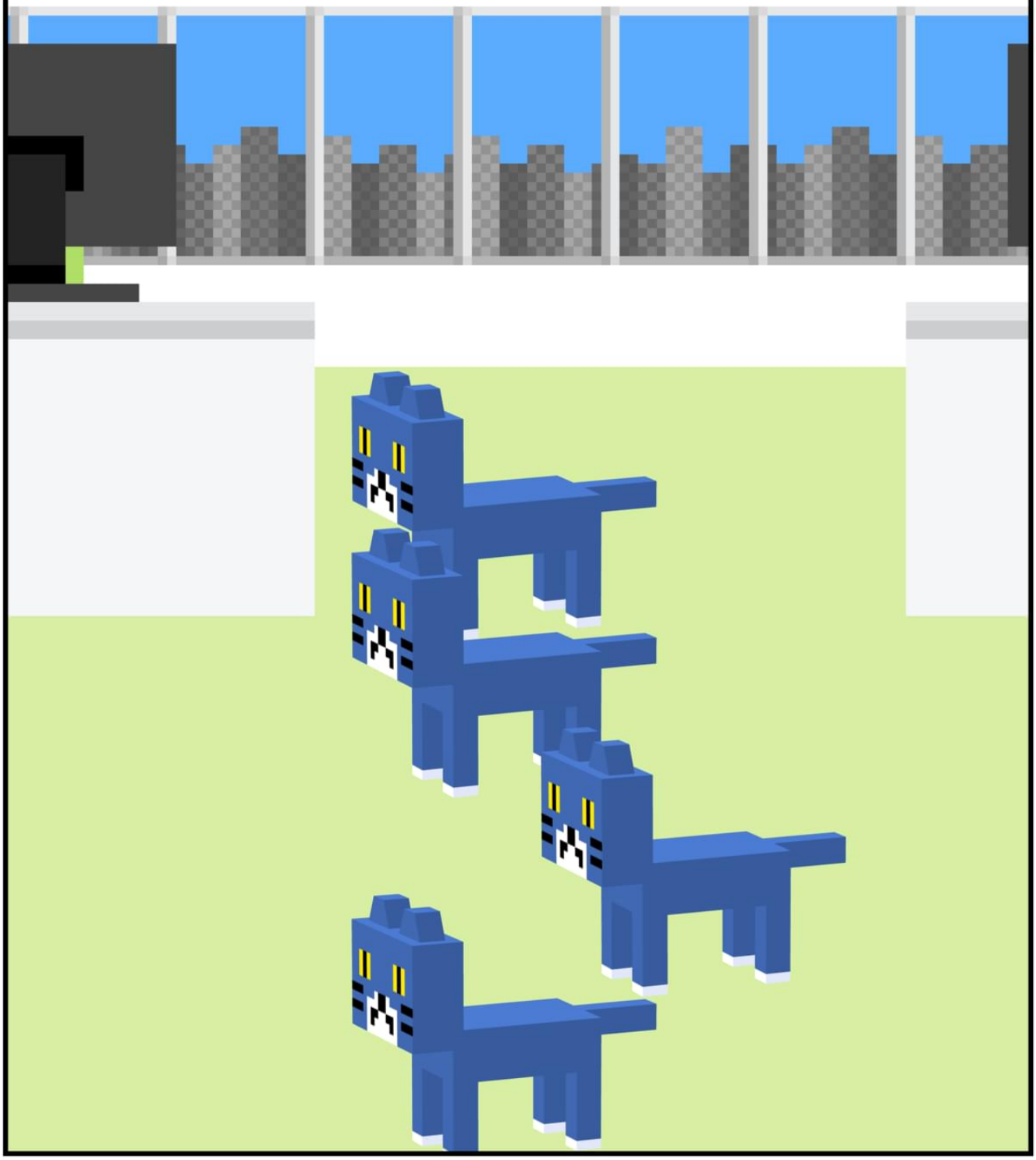


第3章

基本の基本を知ろう



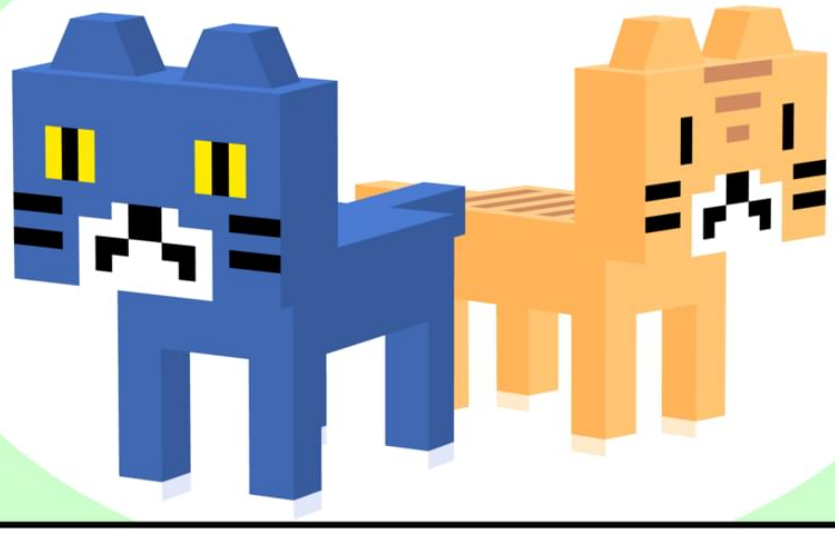
3-1「処理順とインデント」



忠商企業　開発室

青太郎と
虎次郎

ニャー



開発室長　猫野美実（ねこのみみ）

新入社員　新見入子（にいみいりこ）

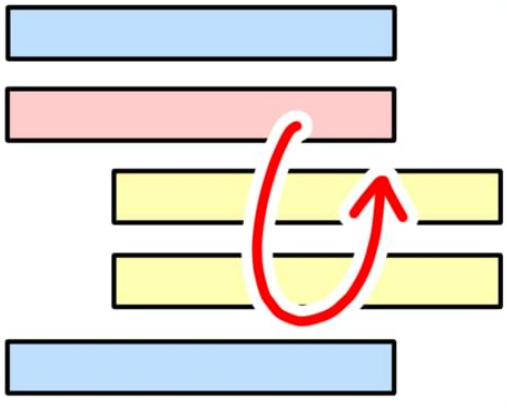

今回からは
Ｐｙｔｈｏｎの
プログラムの簡単な
見方を教える
基礎の基礎ですね
そうだ
今回学ぶことを
まとめておくぞ
・処理順
・インデント
２つなら
何とかなると
思います
今回は
かなり内容を
絞り込んだ

1行目の処理
2行目の処理
3行目の処理
4行目の処理
⋮

1：前に3歩進め
2：右に向きを変えろ
3：前に2歩進め
4：お辞儀をしろ

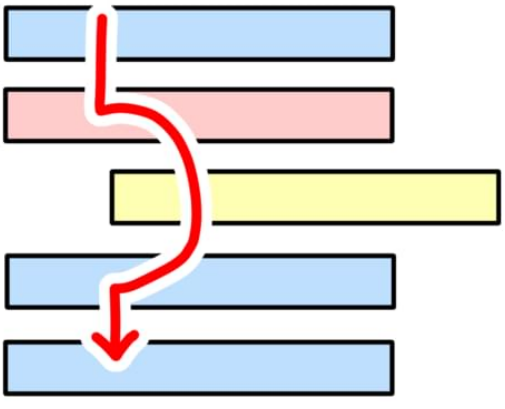
繰り返し処理

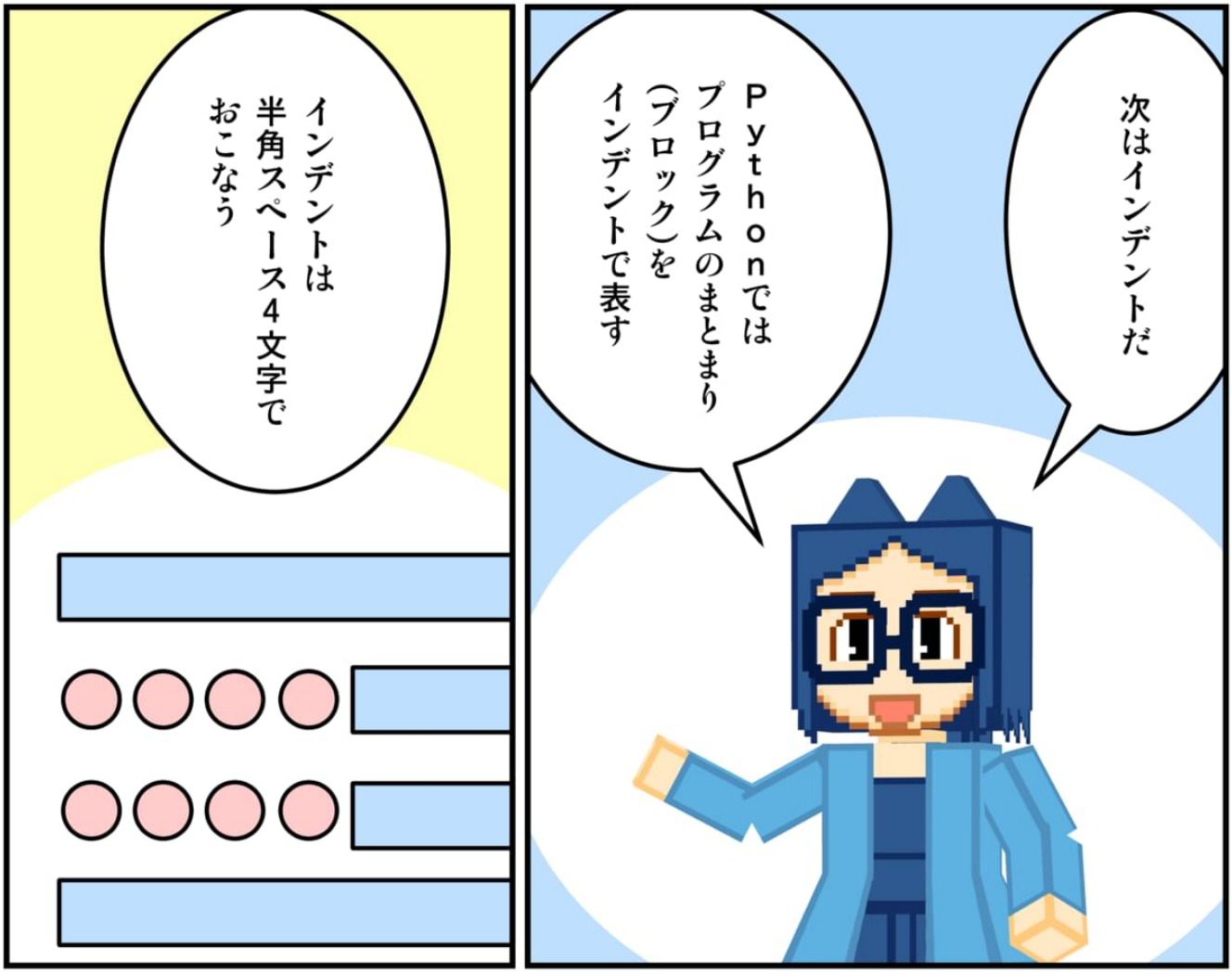
同じ場所を
ぐるぐる繰り返す



条件分岐

条件によって
処理を飛ばす




次はインデントだ
Ｐｙｔｈｏｎでは
プログラムのまとまり
（ブロック）を
インデントで表す
インデントは
半角スペース４文字で
おこなう

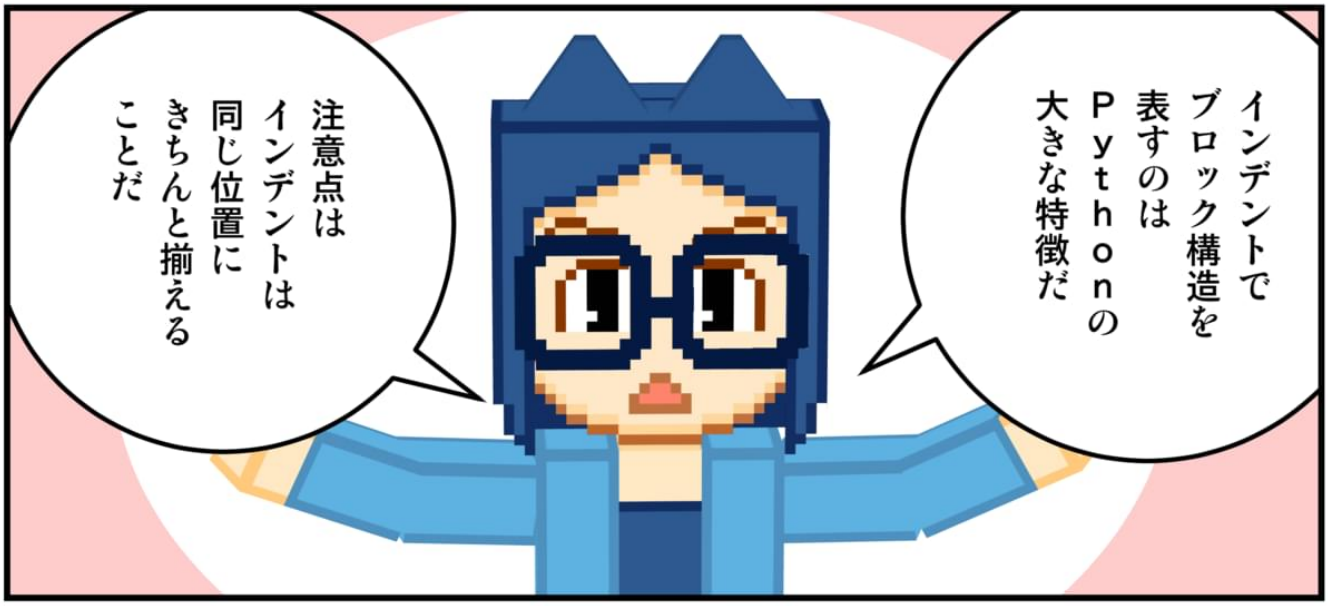

ＶＳＣｏｄｅで
タブキーを押すと
半角スペース４文字で
インデントされる
[Tab]
インデント
[Shift+Tab]
インデント解除

半角スペース4文字でインデント

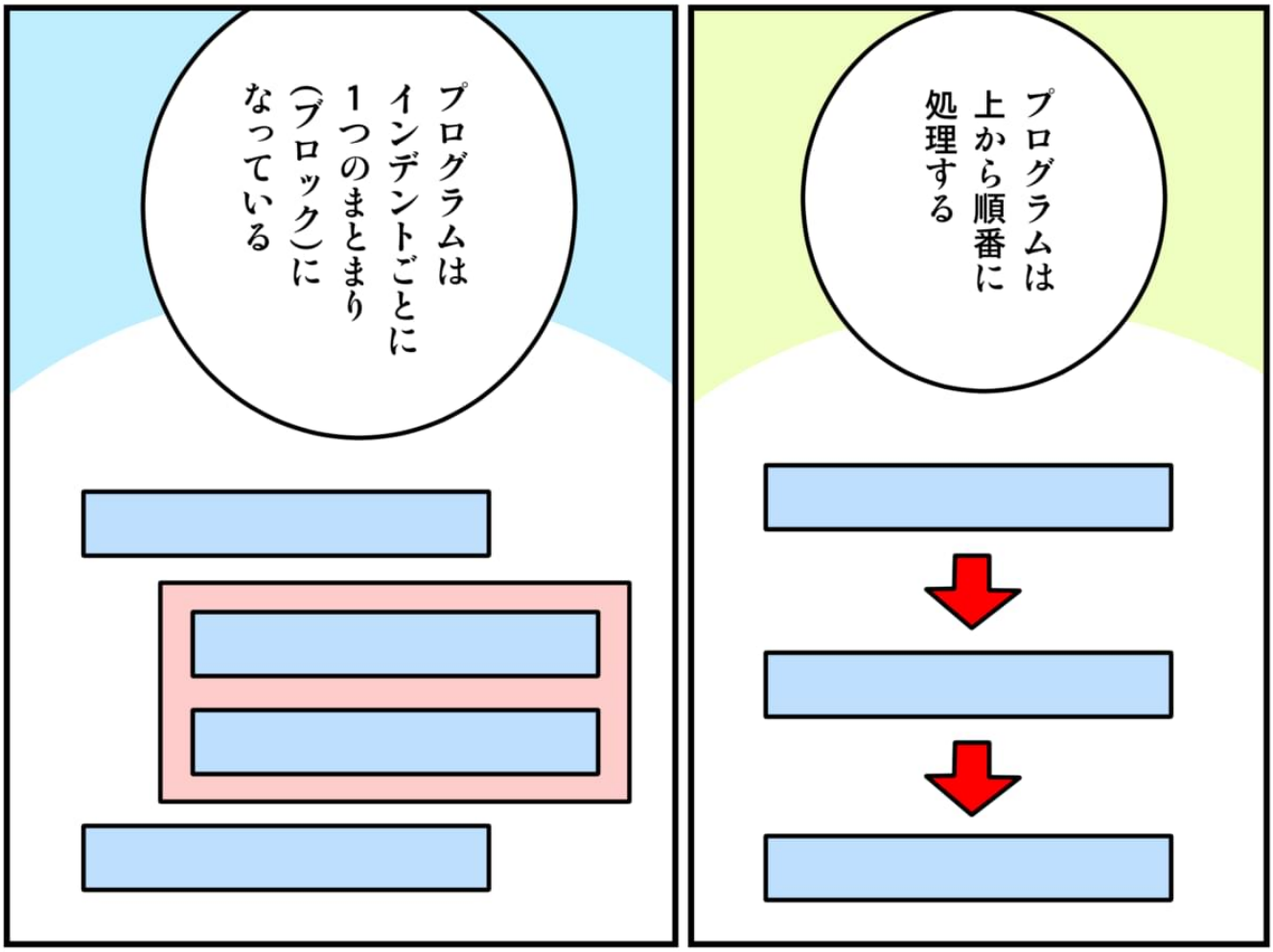
```
program
program
    program
    program
        program
        program
    program
    program
program
program
    program

    program
```

インデントごとに
同じブロック

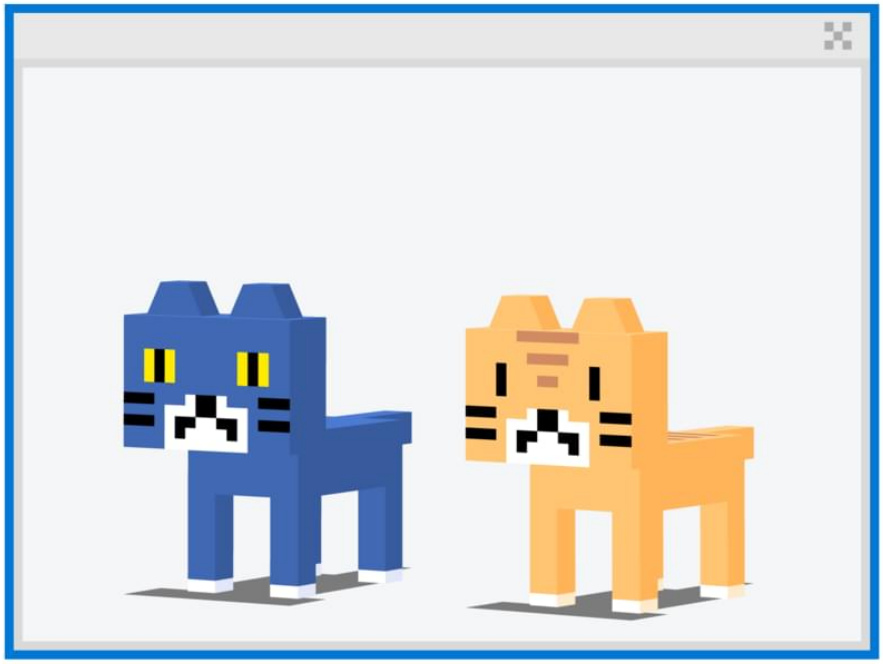
行の間に
空行があっても
同じブロック




それでは
これまで学んだ
ことを私に説明
してくれ
分かりました


プログラムは
上から順番に
処理する
プログラムは
インデントごとに
1つのまとまり
(ブロック)に
なっている


よし、プログラムの
見方の基礎の基礎は
身につけたな
次はさらに細かい
ところを見ていくぞ
分かりました！
頑張っているな
新見くん
にやー



3-2「コメント」


今回はコメントだ
プログラム内の
無視される部分だ
無視されるって
どういうこと
ですか？


人間向けのメモだ
コンピューターは
無視して処理を
おこなうんだ
各行の「#」の右側がコメント
無視される
program
無視される
#を使うのは
Ｐｙｔｈｏｎの
特徴の１つだ

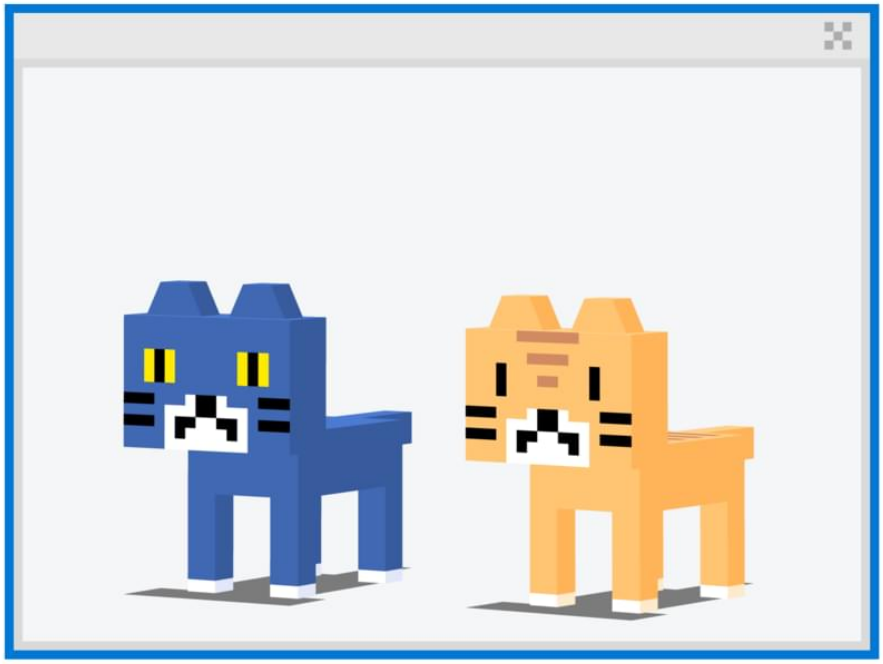

```
# 画像を表示
.....
.....
    # ファイルの読み込み
    .....
    .....
    # 画面に表示
    # ウィンドウより大きい
    # なら縮小して表示
    .....
    .....
```


コメントは
しっかりと多めに
書いた方がよい
他人がパッと見て
処理や意図が分かる
ようにしておくんだ
他人には
将来の自分も
含まれる


将来の自分？
だいたい半年もすれば
内容を忘れている
経験者は語る


あと初心者のうちは
調べた内容や
処理の説明を
細かく書いて
おくとよい
英語を学習
するときに
英文の下に
単語の意味を
書くような
ものだな


なるほど
最初のうちは
たくさん書きます
ものね
というわけで
今後の学習では
コメントを大いに
活用していってくれ



3-3「プログラムの要素1」

そういえば

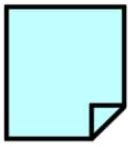
処理の順番とか
まとまりとか
無視される場所とか
習いましたけど

プログラム自体は
まだ習っていない
ですよね

よく気付いた
自分から気付くとは
見所がある

えへへ

まずテキスト
ファイルを作る



main.py

このファイルに
短いプログラムを書く

```
print(1+2)
```

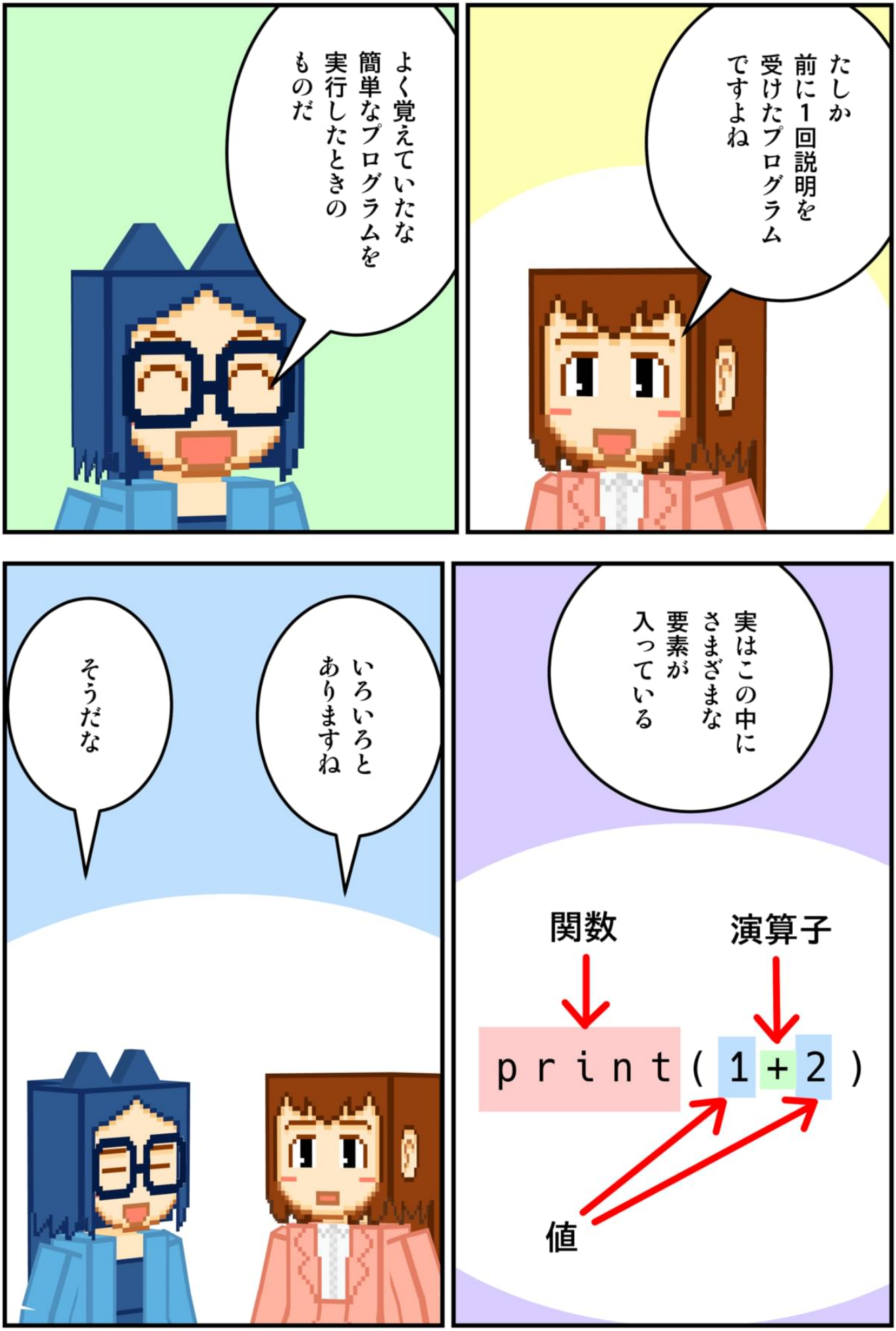
1. VSCodeでmain.pyのある
   フォルダーを開く

2. Ctrl+@でターミナルを開く

3. 次のコマンドを実行
   python test.py

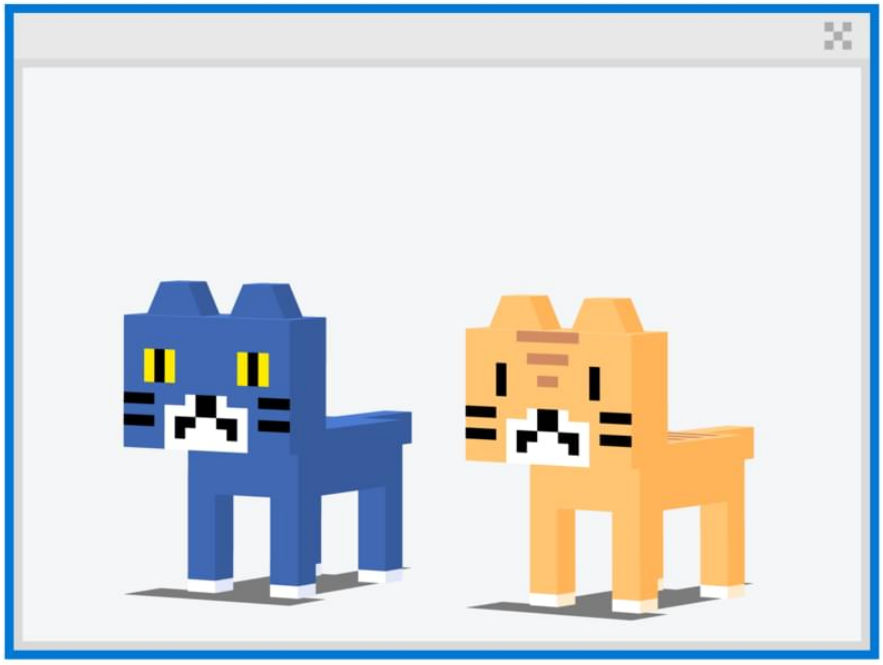
4. 結果表示

```
> 3
```


たしか
前に1回説明を
受けたプログラム
ですよね
よく覚えていたな
簡単なプログラムを
実行したときの
ものだ
実はこの中に
さまざまな
要素が
入っている
関数
演算子
print(1+2)
値
いろいろと
ありますね
そうだな

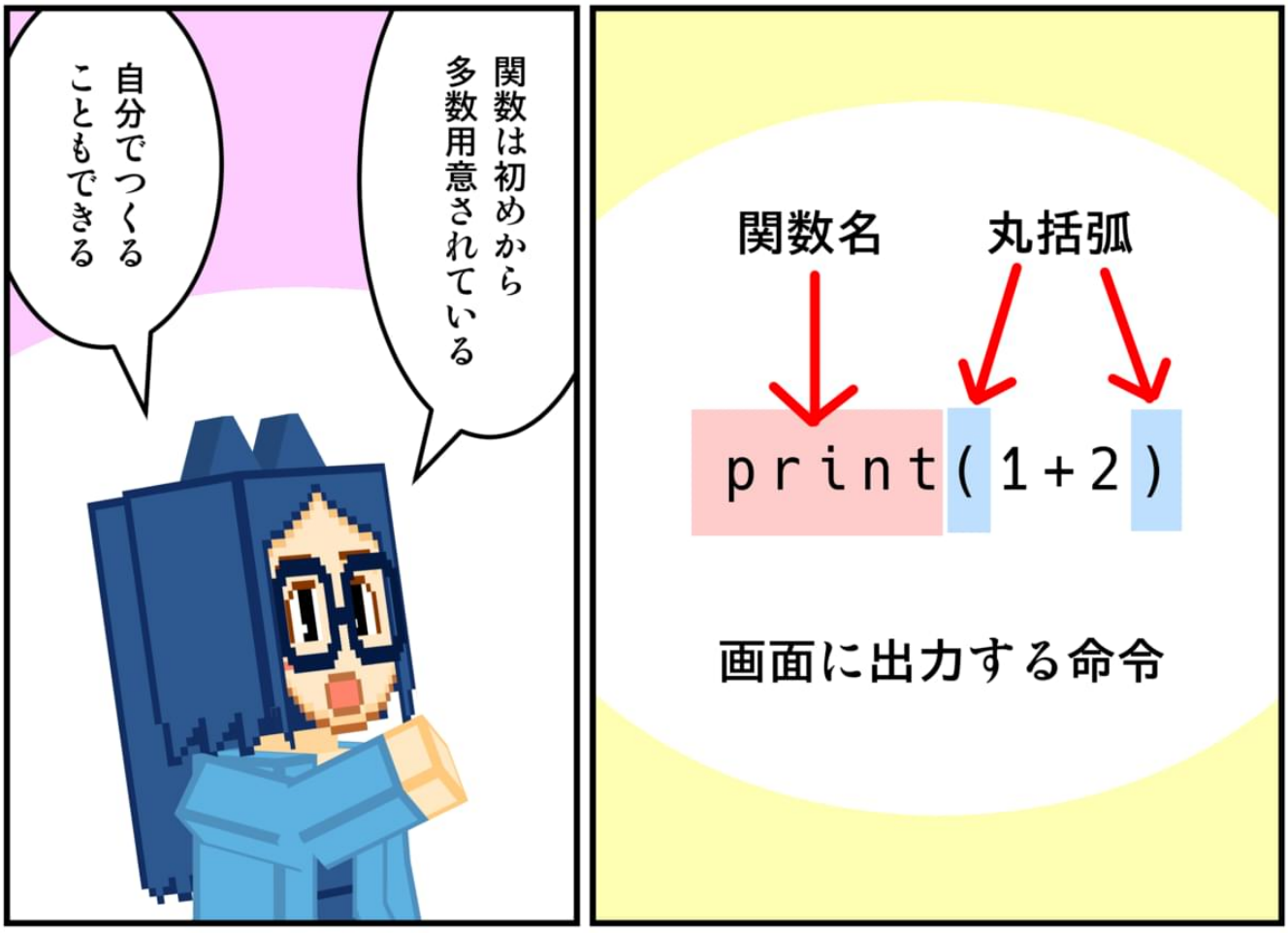

次回からは
関数、値、演算子を
簡単に説明していく
人生で
初めて聞く
内容ですね


プログラミングは
現実世界から
少し離れた
内容になる
からなあ
振り落とされ
ないように
頑張ります！



3-4「関数」


まずは関数だ
関数は
計算したり
処理を実行したり
する命令だ
関数を使う時は
関数名のあとに
丸括弧を書く


関数は初めから
多数用意されている
自分でつくる
こともできる
関数名
丸括弧
print(1+2)
画面に出力する命令

引数1つ

```
print(1)
```

引数なし

```
print()
```

カンマ区切りで複数の引数

```
print(1, 2, 3)
```

例：引数の中で
最も大きな値を
返す関数

```
max(1, 2, 3)
```

↓

```
3
```


```
print(max(1, 2, 3))
```

↓

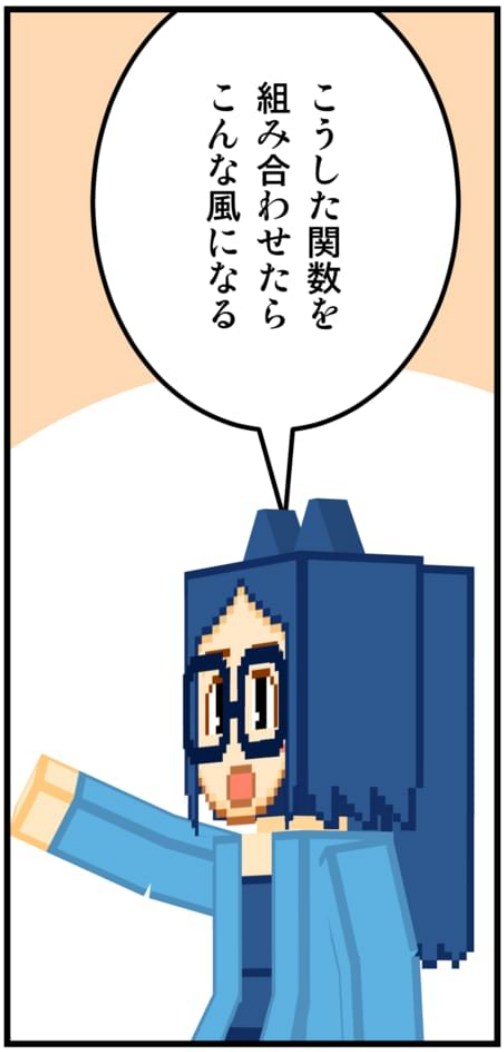
```
max(1, 2, 3)
```
は3

↓

```
print(3)
```
の状態になり

↓

「3」と表示



1. VSCodeでフォルダーを開く
2. フォルダー内に「main.py」を作る
3. 「main.py」にプログラムを書く

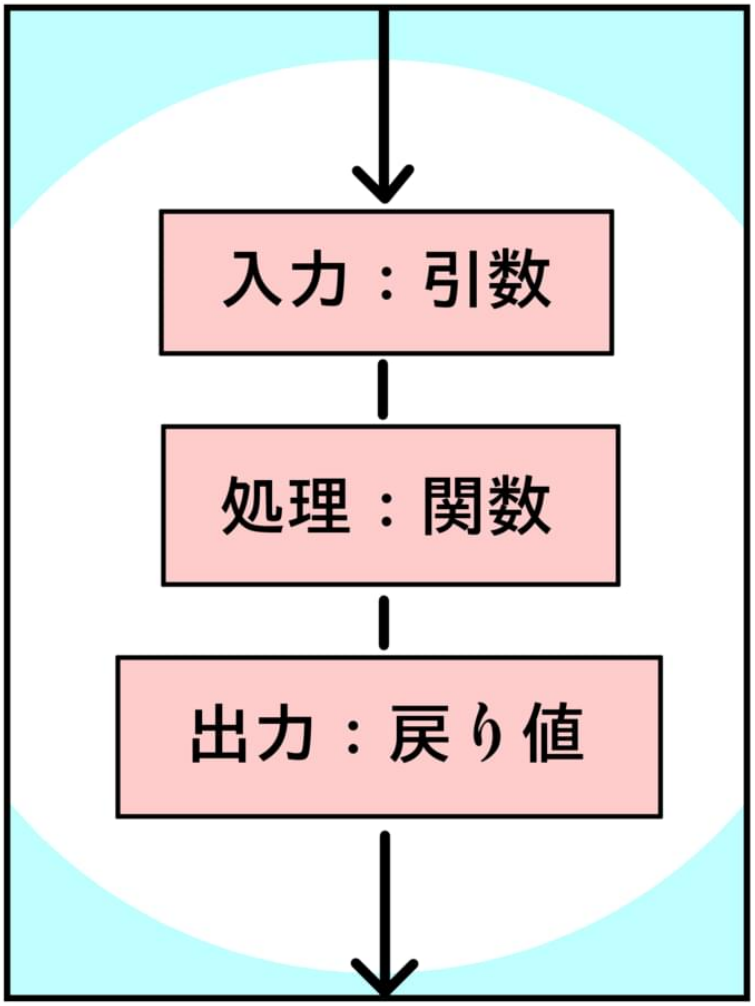
main.py

```
print(max(1, 2, 3))
```

4. Ctrl+@でターミナルを開く
5. 「python main.py」を実行する
6. ターミナルに「3」と表示される

コマンドを実行

```
python main.py
> 3 ←―― と表示
```



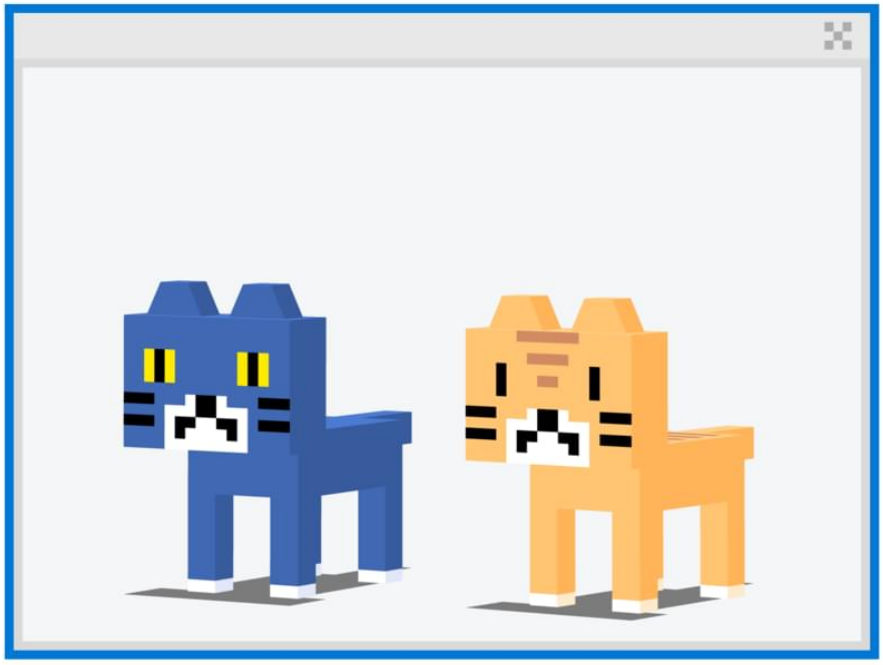
組み込み関数 Python ドキュメント

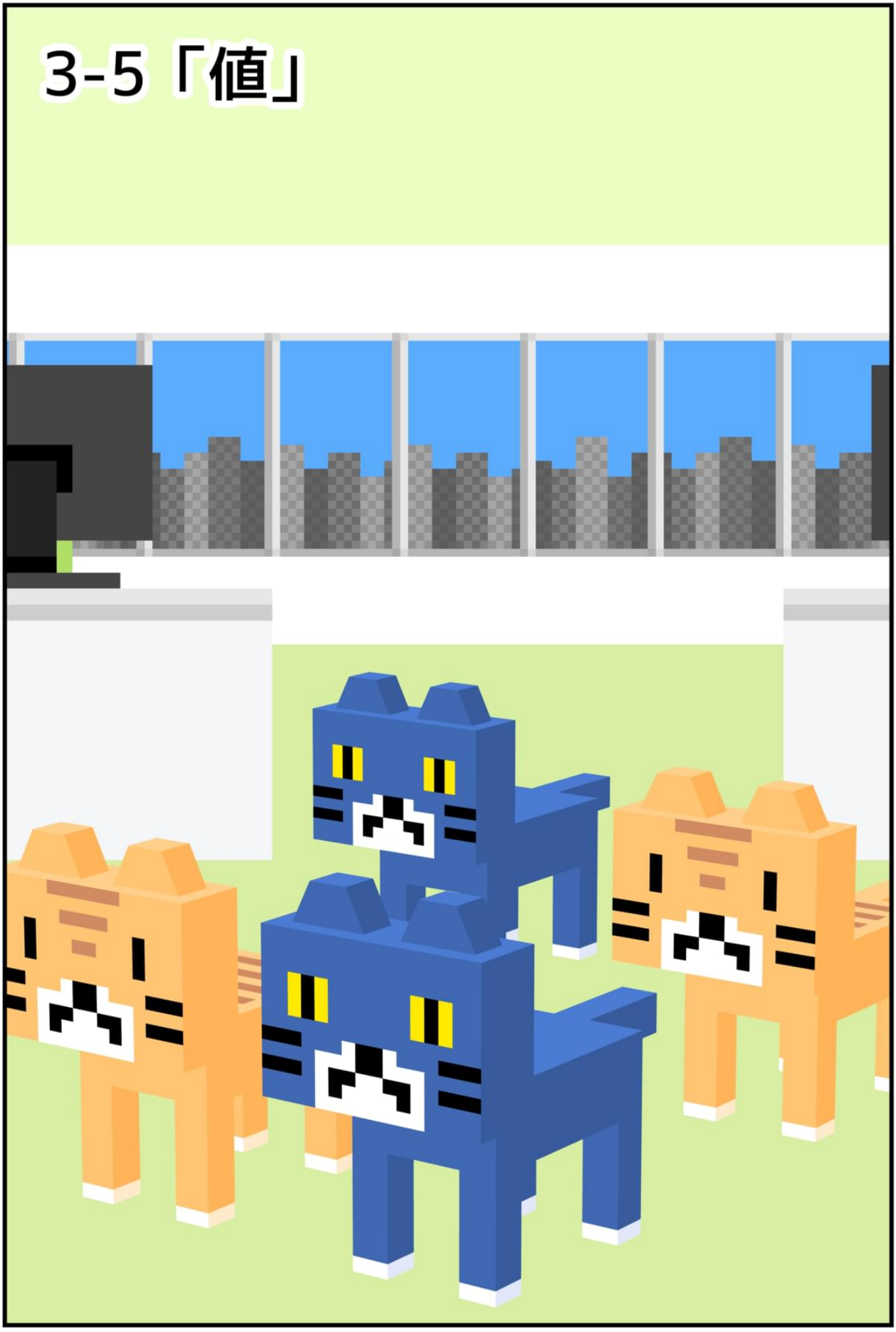
https://docs.python.org/ja/3/library/functions.html

※ 組み込み関数：
初めから入っている関数


プログラムは
関数が重要
なんですね
そうだ
たとえば車を作るときは
小さな部品を組み合わせて
大きな部品にしていくだろう


プログラムでは
自分で関数を作れる
そして
関数を組み合わせたり
まとめた大きな関数を
作ったりしていくんだ
関数の作り方は
あとできちんと
やるからな

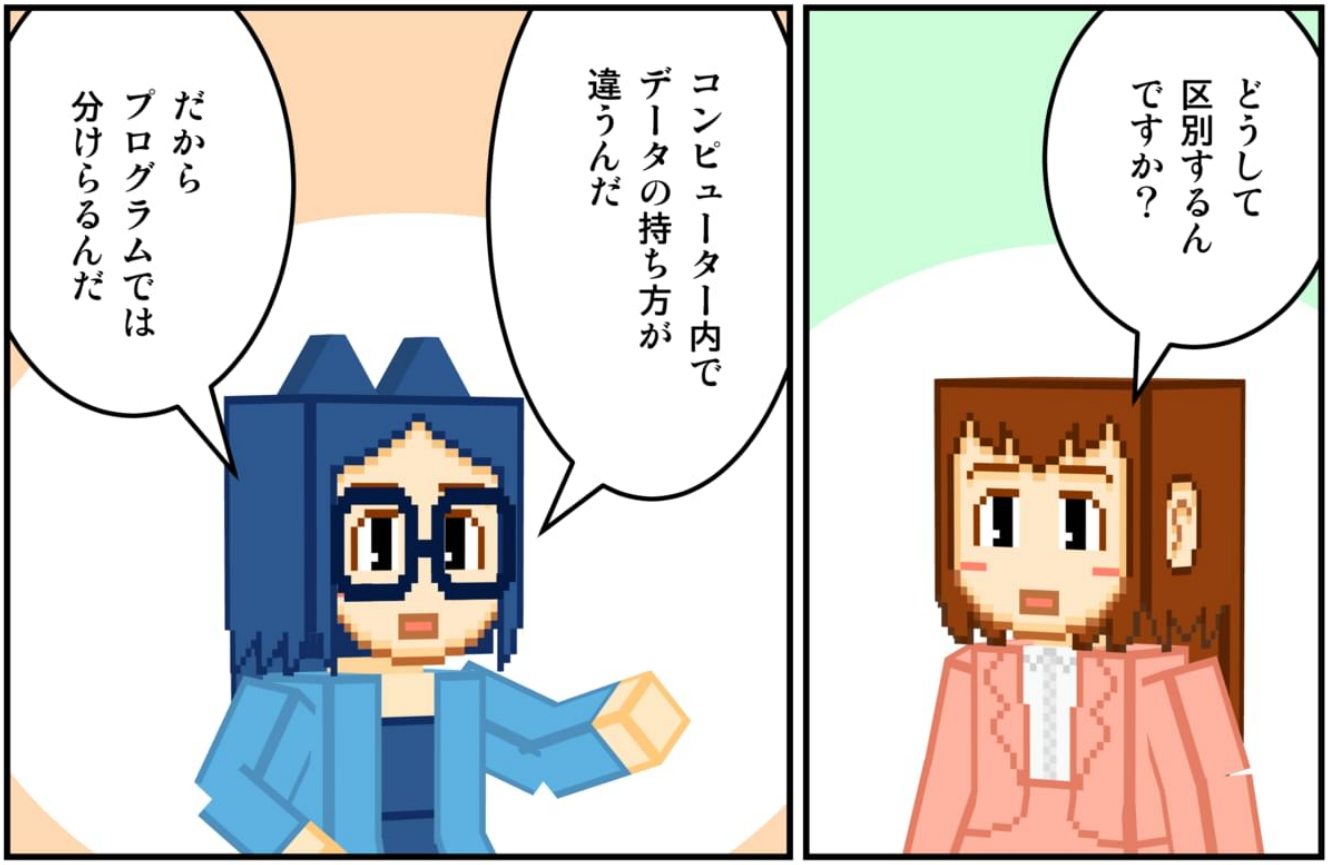



3-5「値」


次は値だ
プログラムには
さまざまな値が
出てくる
数値
1234
テキスト(文字列)
Abcdef
Bool値(真偽値)
True か False
他にもいろいろある


いろいろ
ありますね
プログラムでは
さまざまな値を
使い分けるんだ
もう少しだけ
詳しく説明して
いくぞ


まずは数値だ
数値には整数と
小数点数があり
この2つは区別される
数値
整数
1234
0
-1234
小数点のつかない数値
小数点数
12.34
0.0
-12.34
小数点のついた数値


どうして
区別するん
ですか？
コンピューター内で
データの持ち方が
違うんだ
だから
プログラムでは
分けらるんだ

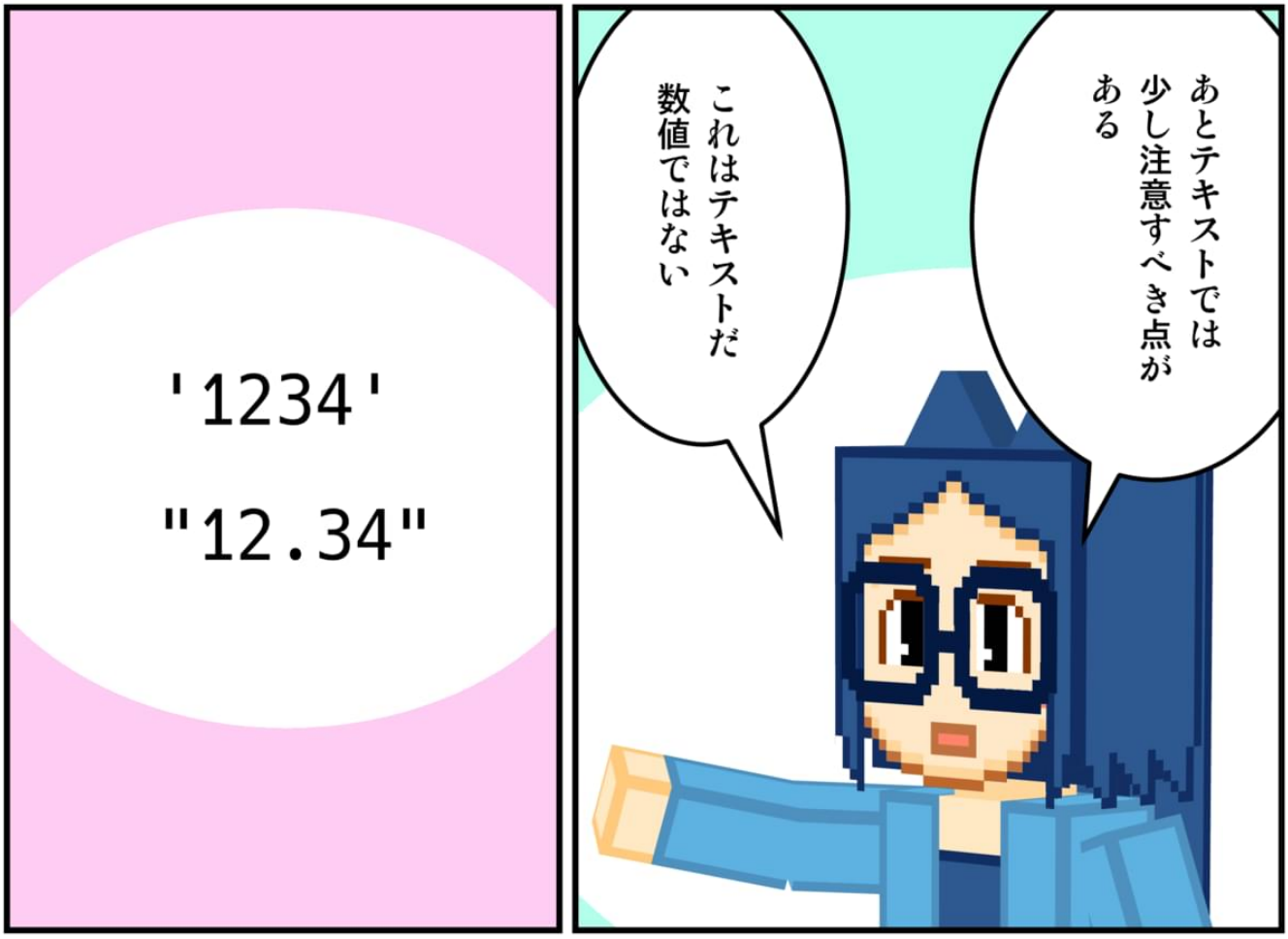
テキスト(文字列)

```
'My Cat.'
"Your Cat!"
'A'
"z"
```

シングルクォーテーションか
ダブルクォーテーションで囲う

中身がないのもあり
空文字と呼ぶこともある

```
''
""
```


あとテキストでは
少し注意すべき点が
ある
これはテキストだ
数値ではない
'1234'
"12.34"


どうしてですか？
クォーテーションで
囲っているからな
数値とは
みなされない

改行は\n

```
'猫の名は\n虎次郎'
```

バックスラッシュ
自体は\\

```
'仕事\\文書.txt'
```

テキスト内の
クォーテーションは
\を付ける

```
'大切なのは\'心\'です'
"大切なのは\"心\"です"
```

タブ文字は\t

```
'動物\t猫\t虎次郎'
```

Bool値(真偽値)

True

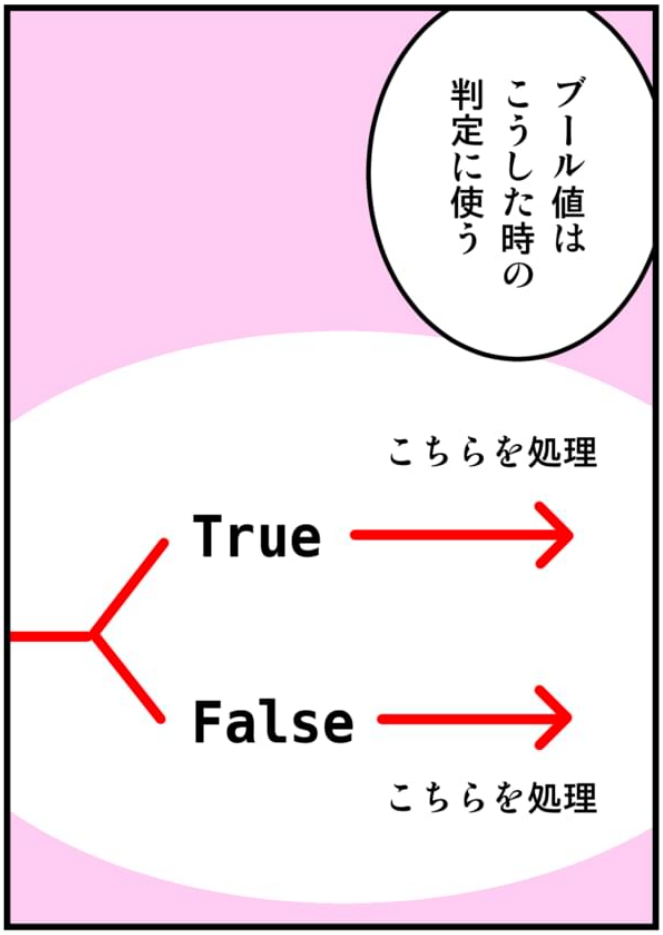
真

はい、OK、有効などの意味

False

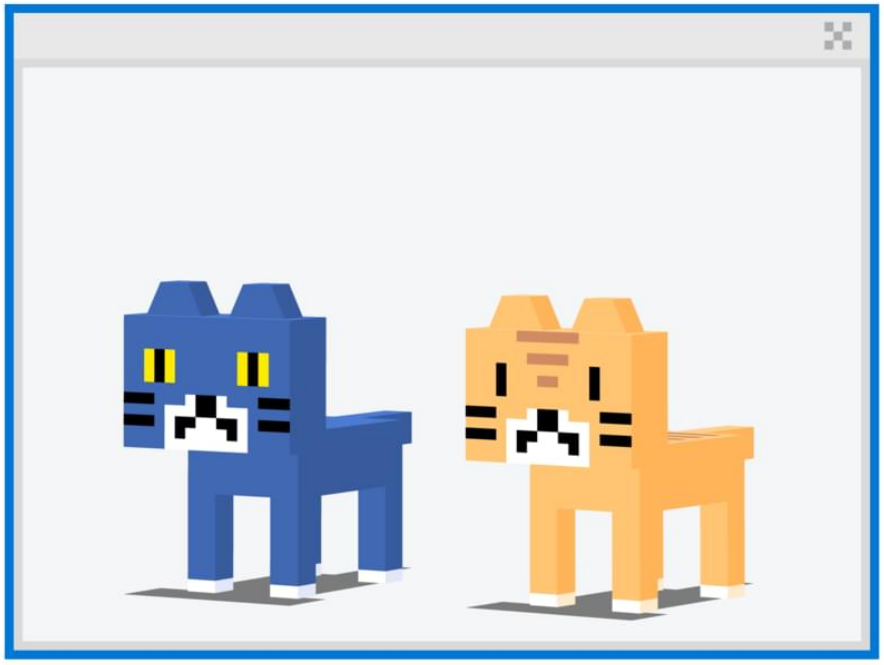
偽

いいえ、No、無効などの意味




というわけで
今回はこんな
ところだな
ありがとう
ございます！


じゃあ、会社猫と
たわむれていいぞ
えへへ
先は長いなあ



3-6「演算子」