

# Google Colaboratory(Colab)チュートリアル

## 1. Google Colab とは

Google Colab(正式名称「Colaboratory」)は、Python をブラウザ上で記述・実行できるサービスです。Colab は学生をはじめ、データサイエンティストや AI 開発者まで幅広く利用されています。ブラウザのみで使用できるため、インストールや環境構築は不要です。

## 2. セットアップ

Colab を利用するには Google アカウントにログインした状態で以下の[公式サイト](https://colab.research.google.com/)にアクセスします。

<https://colab.research.google.com/>



アクセスすると公式チュートリアルが表示されます。ここから新規のノートブックを作成しましょう。ノートブックとはコードやテキストをまとめる形式です。「ファイル」->「ノートブックの新規作成」をクリックします。新規のノートブックが作成されます。ノートブックは Google Drive に保存されます。

### 3. コードを実行する

Python コードを実行するためにコードセルを追加しましょう。

「+ コード」をクリックするとコードセルがノートブックに追加されます。コードセルにはコード実行ボタンとコードを記述する場所が含まれています。



コードセルに Python コードを記述し、実行してみましょう。

「コード実行ボタン」または「ランタイム」->「現在のセルを実行」をクリックしましょう。実行結果がコードセル下部に出力されます。



## 4. Python コードの使用例

このトピックでは実際の Python コードの使用例を紹介します。限定的な内容となりますので、Colab の使い方の参考としてください。

### 4.1. 四則演算の例



The screenshot shows a Google Colab notebook titled "Untitled0.ipynb". The interface includes a top menu bar with options like "ファイル", "編集", "表示", "挿入", "ランタイム", "ツール", "ヘルプ", and "すべての変更:". Below the menu, there are buttons for "+ コード" and "+ テキスト". The main area contains a code cell with the following Python code:

```
print(10 + 20)
print(10 - 20)
print(10 * 20)
print(10 / 20)
```

The output of the code cell is displayed below the code:

```
30
-10
200
0.5
```

### 4.2. 変数の例



The screenshot shows a Google Colab notebook titled "Untitled0.ipynb". The interface is similar to the previous one. The code cell contains the following Python code:

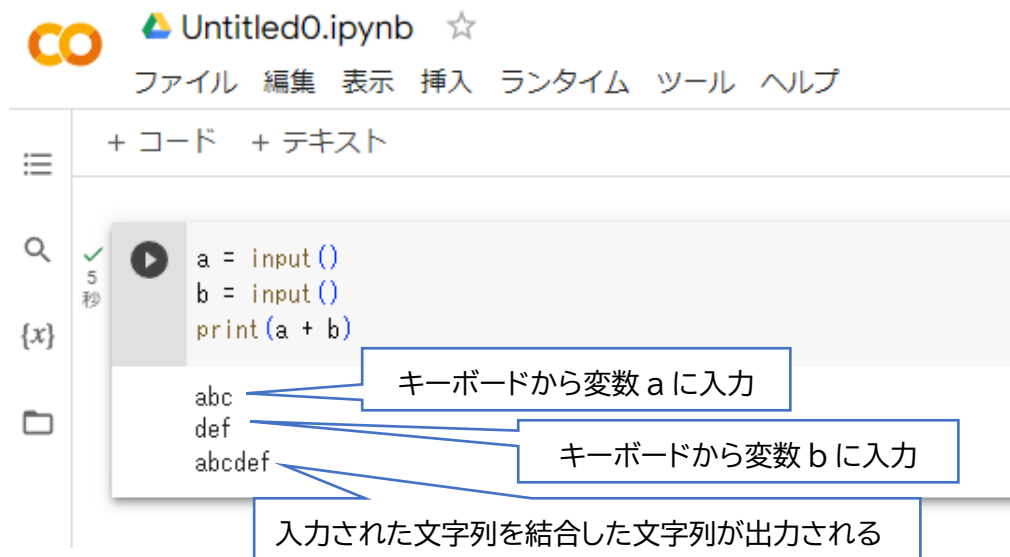
```
a = 12345
b = 67890
c = a + b
print(c)

d = "abc"
e = "def"
f = d + e
print(f)
```

The output of the code cell is displayed below the code:

```
80235
abcdef
```

### 4.3. 入力の例



CO Untitled0.ipynb ☆  
ファイル 編集 表示 挿入 ランタイム ツール ヘルプ

+ コード + テキスト

```
a = input()  
b = input()  
print(a + b)
```

5 秒

abc  
def  
abcdef

キーボードから変数 a に入力

キーボードから変数 b に入力

入力された文字列を結合した文字列が出力される

### 4.4. キャストの例



CO Untitled0.ipynb ☆  
ファイル 編集 表示 挿入 ランタイム ツール ヘルプ すべての変更を保存しました

+ コード + テキスト

```
a = int("10")  
print(a + 10)  
  
b = int("abc") # エラーが出力される  
print(b + 10)
```

0 秒

20

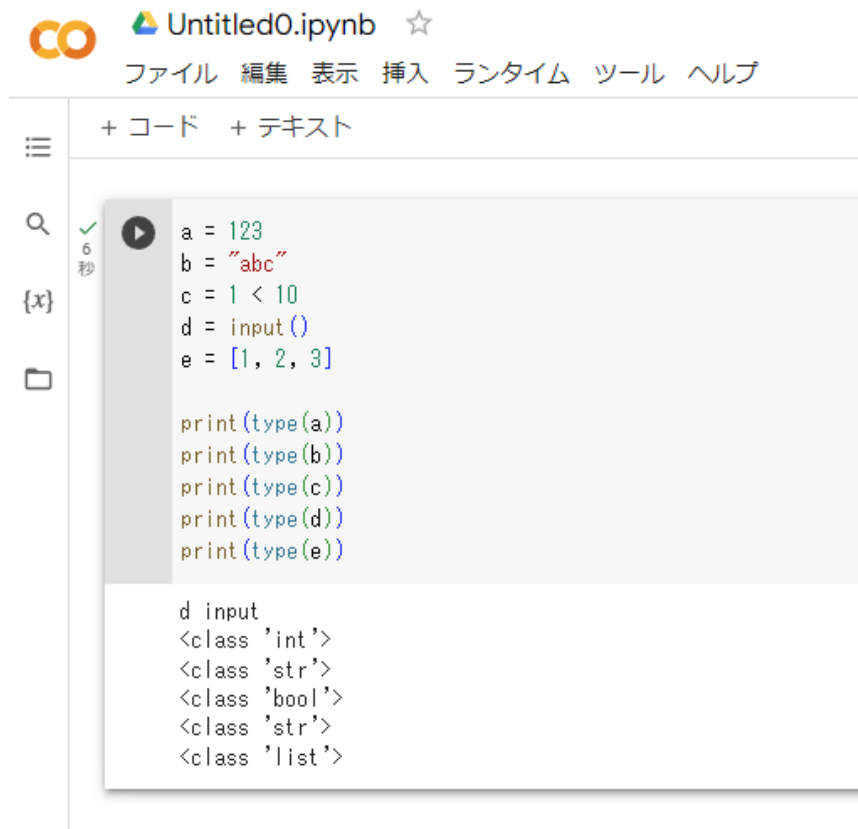
-----

ValueError Traceback (most recent call last)  
<ipython-input-37-6cf179b4ec9c> in <cell line: 4>()  
2 print(a + 10)  
3  
----> 4 b = int("abc") # エラーが出力される  
5 print(b + 10)

ValueError: invalid literal for int() with base 10: 'abc'

SEARCH STACK OVERFLOW

## 4.5. データ型確認の例



The screenshot shows a Jupyter Notebook window titled "Untitled0.ipynb". The interface includes a top menu bar with "ファイル", "編集", "表示", "挿入", "ランタイム", "ツール", and "ヘルプ". Below the menu, there are tabs for "+ コード" and "+ テキスト". The main area contains a code cell with a play button icon, a checkmark, and a "6 秒" (6 seconds) execution time. The code in the cell is:

```
a = 123
b = "abc"
c = 1 < 10
d = input()
e = [1, 2, 3]

print(type(a))
print(type(b))
print(type(c))
print(type(d))
print(type(e))
```

The output of the code cell is:

```
d input
<class 'int'>
<class 'str'>
<class 'bool'>
<class 'str'>
<class 'list'>
```

## 4.6. 文字列操作の例



The screenshot shows a Jupyter Notebook window titled "Untitled0.ipynb". The interface includes a top menu bar with "ファイル", "編集", "表示", "挿入", "ランタイム", "ツール", "ヘルプ", and "すべての変更を保存". Below the menu, there are tabs for "+ コード" and "+ テキスト". The main area contains a code cell with a play button icon, a checkmark, and a "0 秒" (0 seconds) execution time. The code in the cell is:

```
print(ord("A")) # アスキーコードの確認
print(len("abcdef")) # 文字列の長さを確認

a = "123"
print(a.isdigit()) # 文字列が数字のみで構成されてるか確認

b = "abc"
print(b.isalpha()) # 文字列がアルファベットのみで構成されてるか確認
```

The output of the code cell is:

```
65
6
True
True
```