通信・計算機システムと知的情報処理 (0/4)

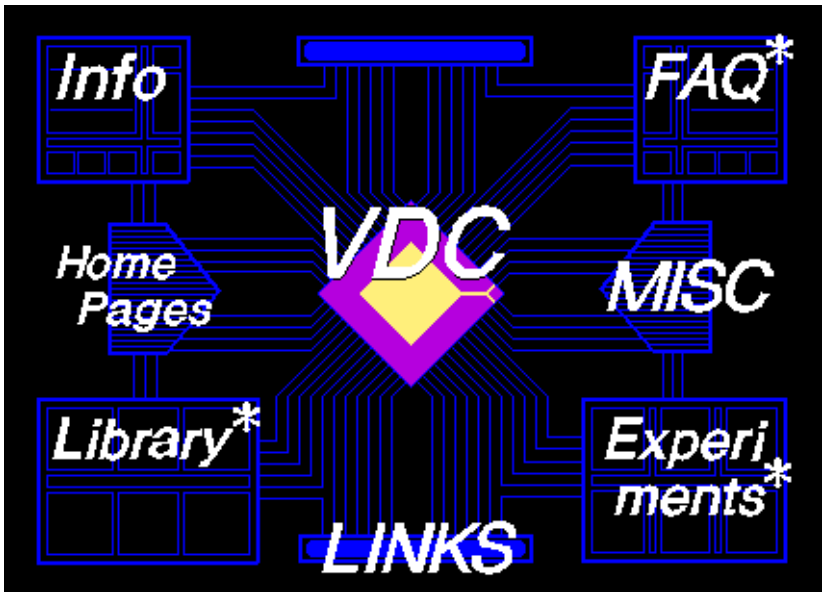
情報通信系

工学リテラシー

資料作成
篠崎、中原、上垣外
およびTAメンバー

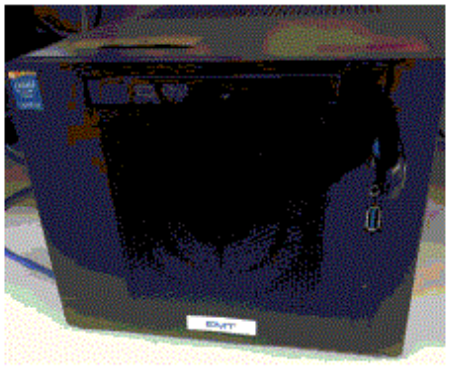
VLSI設計室の利用について

- 土足厳禁、下足入れ番号とスリッパ番号は一致
- 本講義の時間外も、他の講義の時間外は利用可
- 利用については、VLSI設計室のルールに従うこと
  - http://www.vdc.ict.e.titech.ac.jp/index.html



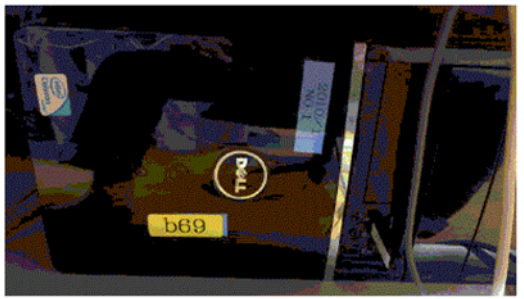
VLSI設計室の計算機の起動方法

- 「工学リテラシ」で利用する計算機は机の下に本体
  - 利用にはアカウントが必要
    - 学生証を見せて、アカウント名とパスワードを受け取る
  - Linux(ネットワークブート)を用いる
  - いくつかの機種が混在



ASUSのPC

DELLのPC



DELLのPC

- 自分のノートパソコン等を使用してもよい
  - キャンパス無線LANに接続し、東工大ポータルへのアクセスが必要
  - win/mac/linuxいずれでもよい
  - サポートはできないので自己責任で使用すること

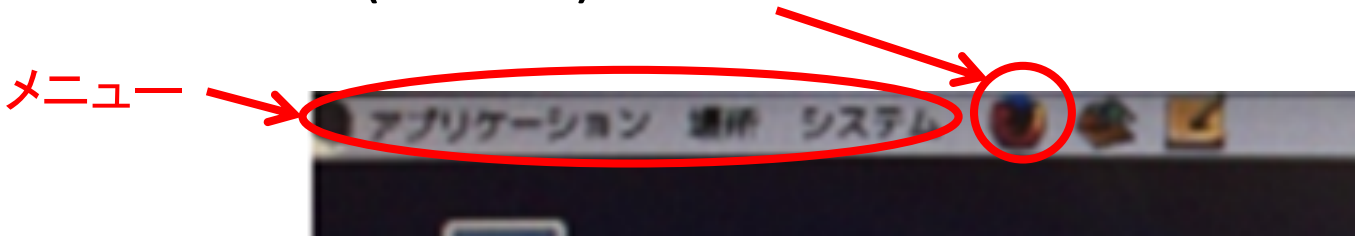
VLSI設計室の計算機の終了方法

1. 画面左上の「システム」を選択（左クリック）
2. 「ログアウト」を選択（左クリック）してログアウト
3. 電源ボタンを押して計算機をシャットダウン

Logout: Click "System" at the top left of the screen, and then click "Logout"

ブラウザ，端末の起動方法

- ブラウザの起動方法
  - 画面左上のメニューから選ぶ（左クリック）、または
  - ブラウザ(Firefox)アイコンを左クリック



- Linux「端末」の起動方法
  - 画面左上のメニューの中から選ぶ，または
  - デスクトップ上で右クリック
    - 表示されるメニューから「端末」を選ぶ
- PDFファイルの閲覧方法
  - Evince (Document Viewer)を起動

VLSI設計室での講義資料の見方

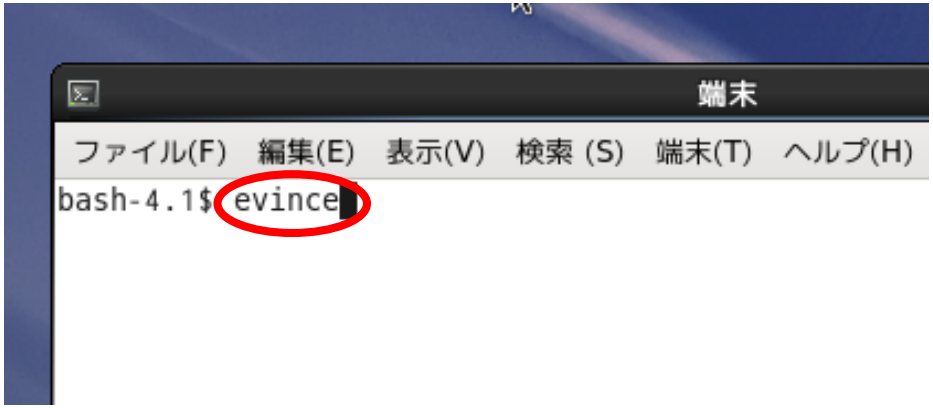
- OCWi またはOCWで本講義のページを開く
- OCW/OCWiから参照したいPDFをダウンロード

OCWi

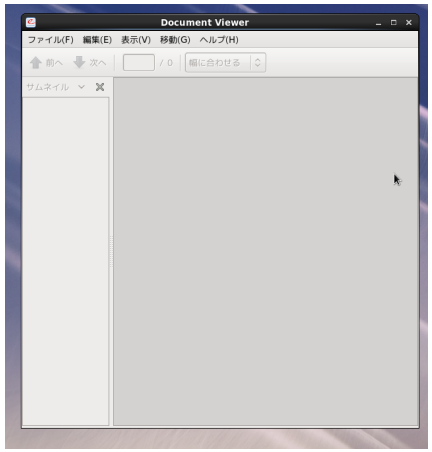
OCW

Evince (Document Viewer)の起動

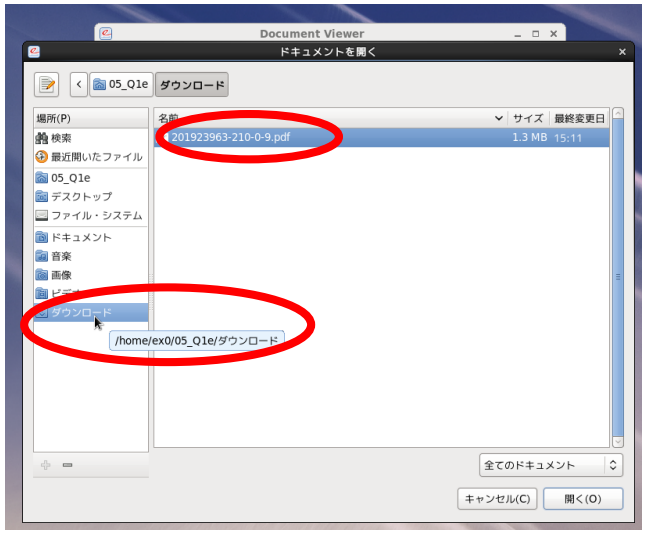
① 「端末」でevinceと入力、Enterキーを押す



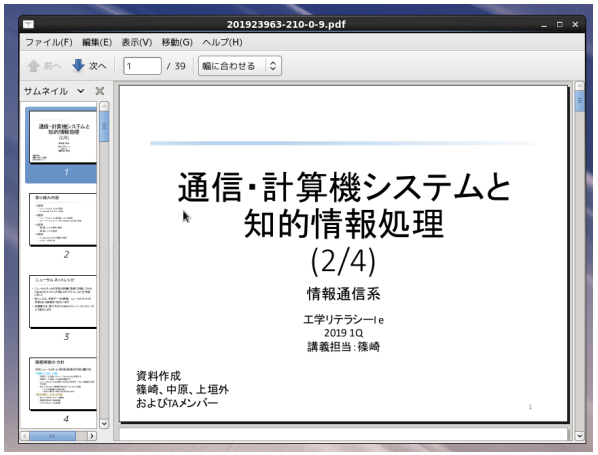
② Evince (Document Viewer)が起動



③ ファイル（PDF）を選択

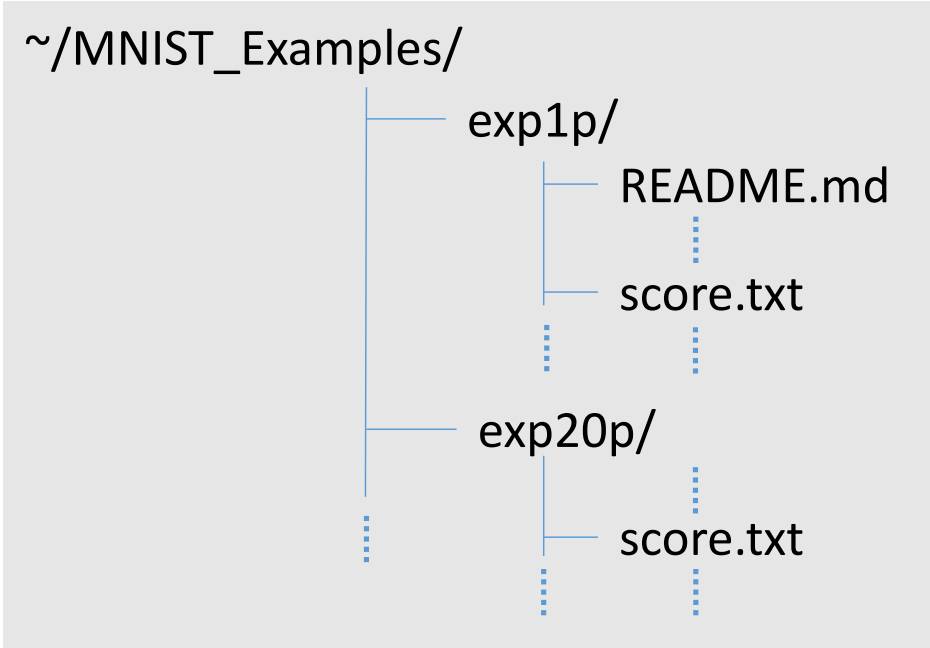


④ ファイル（PDF）が開く



Linux (Unix系OSカーネル)

- コマンド(命令)を入力
- 階層型ディレクトリ構造(ディレクトリ=フォルダ)
  - ルートディレクトリ /
  - ホームディレクトリ ~/
    - 最初の作業ディレクトリ
  - カレントディレクトリ ./
    - 現在の作業ディレクトリ
  - 親ディレクトリ ../
- ディレクトリの指定
  - ルートからの絶対パス
  - カレントディレクトリからの相対パス
- プロンプト(コマンドの入力位置を示す)
  - カレントディレクトリの表示
    - 設定よって表示は異なる




```
19M1[redacted]1@login1:~> cd MNIST_Examples/exp1p
19M1[redacted]1@login1:~/MNIST_Examples/exp1p> ls
README.md    mnist.model      runtsubame.sh           train.sh
eval.log     model            runtsubame.sh.e4716010  train.time.log
eval.py      prep.py          runtsubame.sh.o4716010  train_on_GoogleCoLab.ipynb
eval.sh      prep.sh          score.txt
mkgraph.py   requirement.txt  train.log
mkgraph.sh   run.sh           train.py
19M1[redacted]1@login1:~/MNIST_Examples/exp1p>
```

カレントディレクト

Linuxの基本的なコマンド1

- ls　（ディレクトリ名、ファイル名の表示）(list)
  - $ ls　：カレントディレクトリの表示
  - $ ls␣~/　：ホームディレクトリの表示
  - $ ls␣..　：親ディレクトリの表示
  - $ ls␣path　：pathで指定のディレクトリの表示
    - ✓ pathはルートからの絶対パス、または、カレントディレクトリからの相対パス
- echo（表示）
  - $ echo␣text　：textを表示
- pwd （カレントディレクトリのパス表示）(print working directory)
  - $ pwd
- cd　（カレントディレクトリの変更）(change directory)
  - $ cd␣~/　：ホームディレクトリへ
  - $ cd␣..　：親ディレクトリへ
  - $ cd␣path　：pathで指定のディレクトリへ
- mkdir（ディレクトリの作成）(make directory)
  - $ mkdir␣dirname　：dirnameの名前のディレクトリ作成
    - ✓ dirnameで作成するディレクトリ位置も指定可能

スペース（空白）は「␣」で表示
- 引数、オプションの区切り

Linuxの基本的なコマンド2

- cat（ファイルの連結、中身の表示） (concatenate)
  $ cat␣filename : filenameのファイルの中身の表示
  - ✓ filenameでファイルのディレクトリ指定も可能
- less （ファイルの中身の表示）
  $ less␣filename : filenameのファイルの中身の表示
  - ✓ ”q” を押す終了
- cp （ファイルのコピー） (copy)
- mv（ファイルの移動） (move)
  $ cp␣source ␣destination : sourceファイルをdestinationにコピー
  $ mv␣source ␣destination : sourceファイルをdestinationに移動
  - ✓ カレントディレクトリのファイルの場合、ディレクトリ（へのパス）は省略可
  - ✓ ファイル名を変更しない場合、destinationでファイル名は省略可

- リモート計算機の利用
  - ssh （リモート計算機へログイン）(secure shell)
    $ ssh␣ID@login.t3.gsic.titech.ac.jp : TSUBAMEへのログイン
  - scp （リモート計算機からファイルをコピー）(secure copy)
    $ scp␣ID@machine:source ␣destination : リモートマシンのsourceファイルをdestinationにコピー
  - ssh-keygen（秘密鍵公開鍵の生成）
    $ ssh-keygen␣–t␣rsa

Shell（コマンドインタプリタ）の機能

- 「Tab」キー
  - コマンドやファイル名の補完
  - 一意に定まるところまで名前を補完してくれる
- 「矢印」キー
  - コマンド履歴の表示、編集
    - ↑：前の入力コマンド
    - ↓：次の入力コマンド
    - ←、→：カーソル（編集位置）移動
- 文字の削除
  - 「Del」　　：カーソル位置の文字
  - 「Bachspace」：カーソル位置の前の文字

TSUBAMEで用いるコマンド、スクリプト

- git　　(分散型バージョン管理システム)
  $ git␣clone␣URL
- qsub　(TSUBAMEへのジョブの提出(queue submit))
  $ qsub␣script-file-name.sh
  $ qsub␣-g␣tga-egliteracy␣-l␣h_rt=time␣runtsubame.sh
- qstat　(ジョブの状態(queue status))
  $ qstat
- qdel　(TSUBAMEからのジョブの削除(queue delete))
  $ qdel␣job-ID

日本語キーボード
shift-7　: シングルクォート「’」
shift-@ : バッククォート　「`」

✓ 講義で用意したスクリプト

- mkgraph.sh (折れ線グラフの作成)
  $ ./mkgraph.sh␣’XLabel’␣’X列’␣’YLabel’␣’Y列’␣’出力ファイル名’

TSUBAMEでのエディタ (nano)の起動

- 課金グループ(tga-egliteracy)参加後に利用可能
- パスの事前指定
  - 方法１ （設定ファイルで指定．一度行えば，次回以降は不要）

    $ cp␣~/.bashrc␣~/dot.bashrc.bak

    $ cp␣/gs/hs0/tga-egliteracy/etc/dot.bashrc␣~/.bashrc
  - 方法２ （ログアウトまで有効．ログイン後毎回行う）

    $ alias␣nano=/gs/hs0/tga-egliteracy/local/bin/nano
- 実行方法

  $ nano

- パスの事前指定なしで実行（パスを直接指定）

  $ /gs/hs0/tga-egliteracy/local/bin/nano

注意：VLSI設計室のアカウントについて

- 本クォーターの終了後、本講義で使用したVLSI設計室のアカウントおよびユーザーディレクトリは使用できなくなります
- 本クォーター終了後も必要なデータがある場合は、各自でバックアップしてください

Google Colaboratory

Google Colab(Colaboratory)とは

- 機械学習の普及を目的としたGoogleのサービス
- クラウドで提供されるPythonの実行環境
- 環境構築が不要
- 誰でも無料で利用可能
- プログラムを簡単に共有可能
- Googleアカウントが必要

Google Colabの使い方(目次)

新規作成する場合

(Googleにログインした状態で)

1. Colab(https://colab.research.google.com/)にアクセス
2. 新しいノートブックを押す
3. Pythonプログラムを書いて、実行(　)を押す

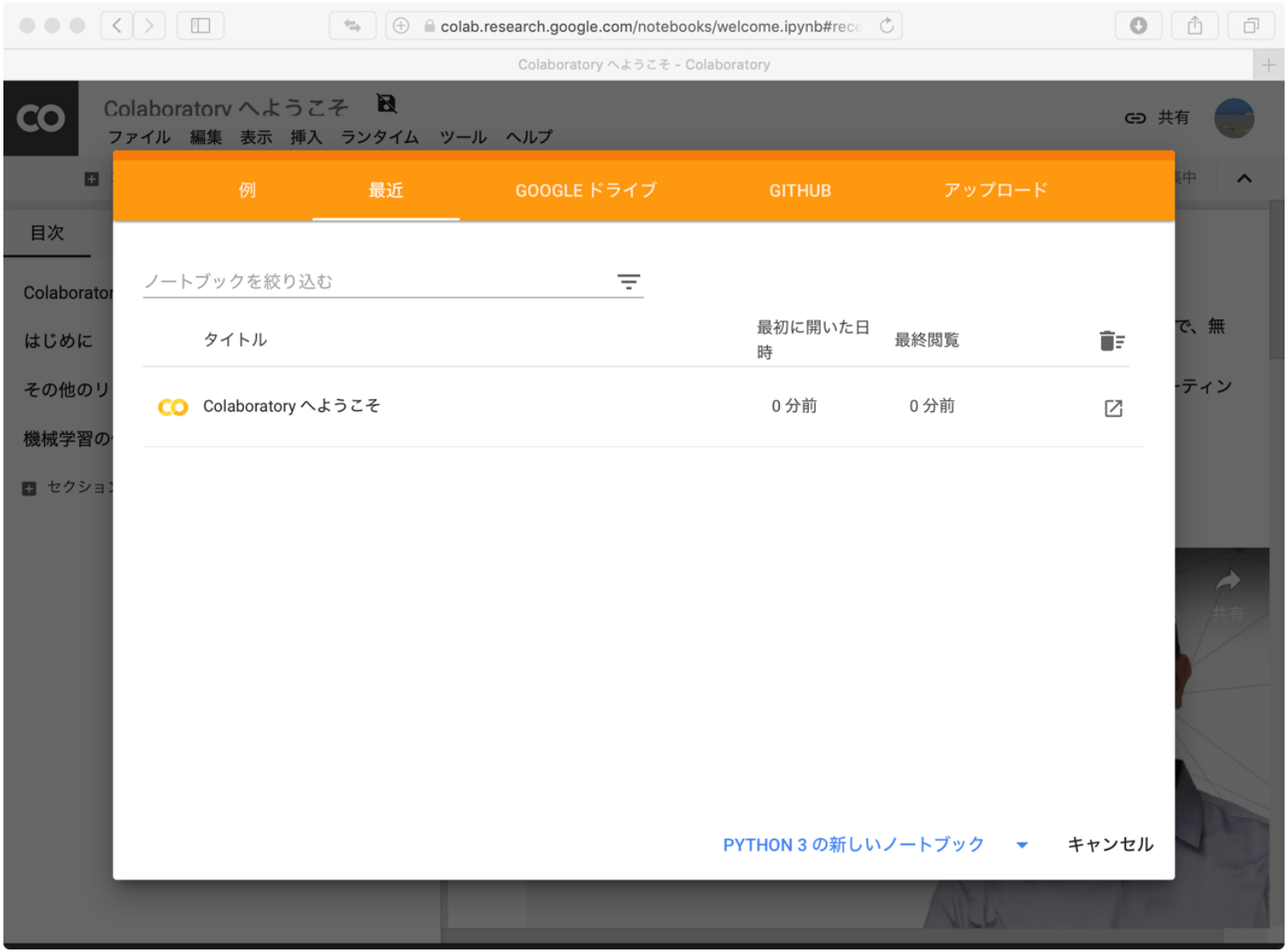
ファイルをGithubから読み込む場合

(Googleにログインした状態で)

1. Colab(https://colab.research.google.com/)にアクセス
2. GithubのレポジトリのURLを入力
3. 読み込むipynbファイルを選択
4. 順番に実行(　)を押す

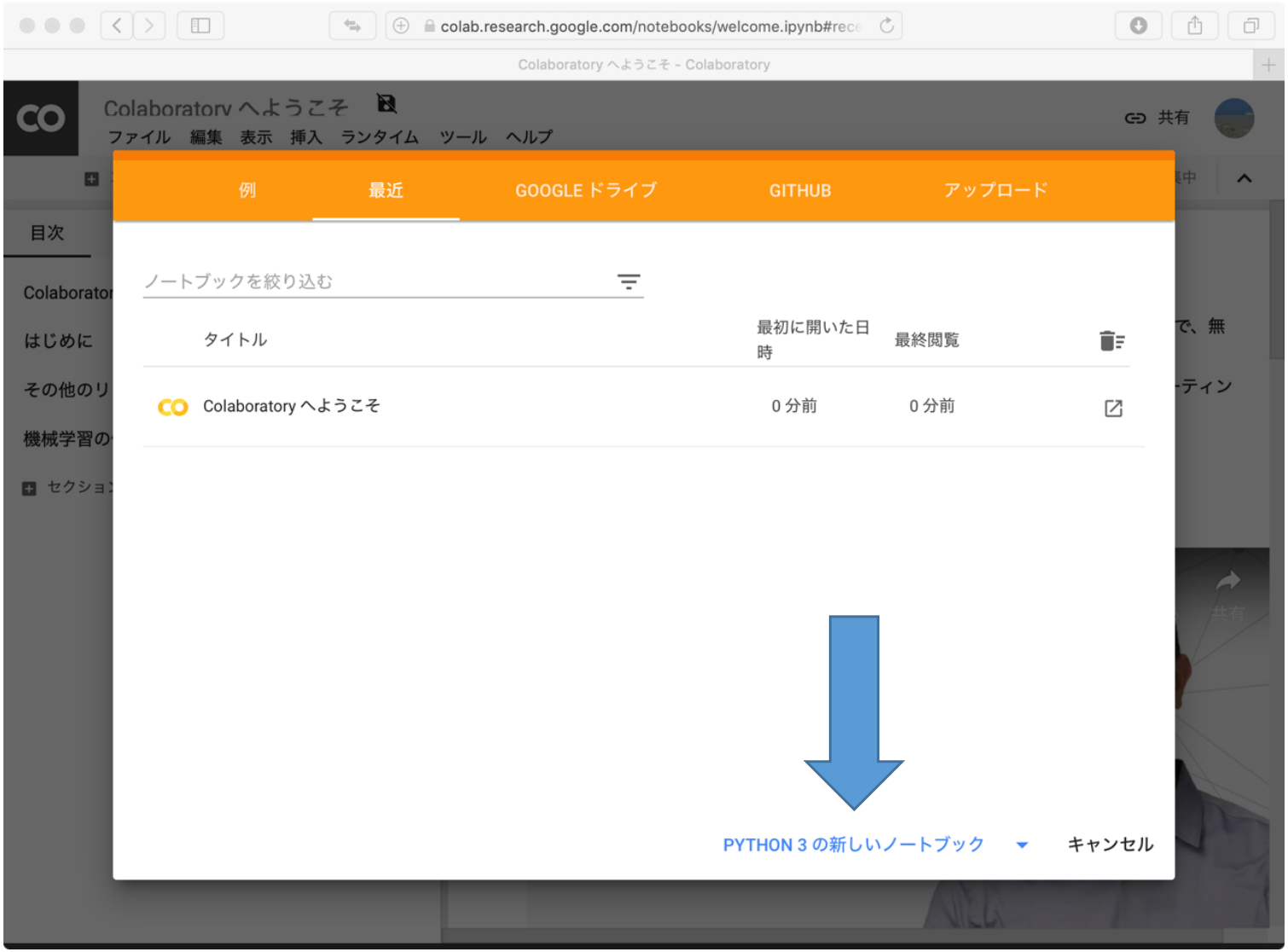
[新規作成] Colabにアクセス

https://colab.research.google.com/ にアクセス



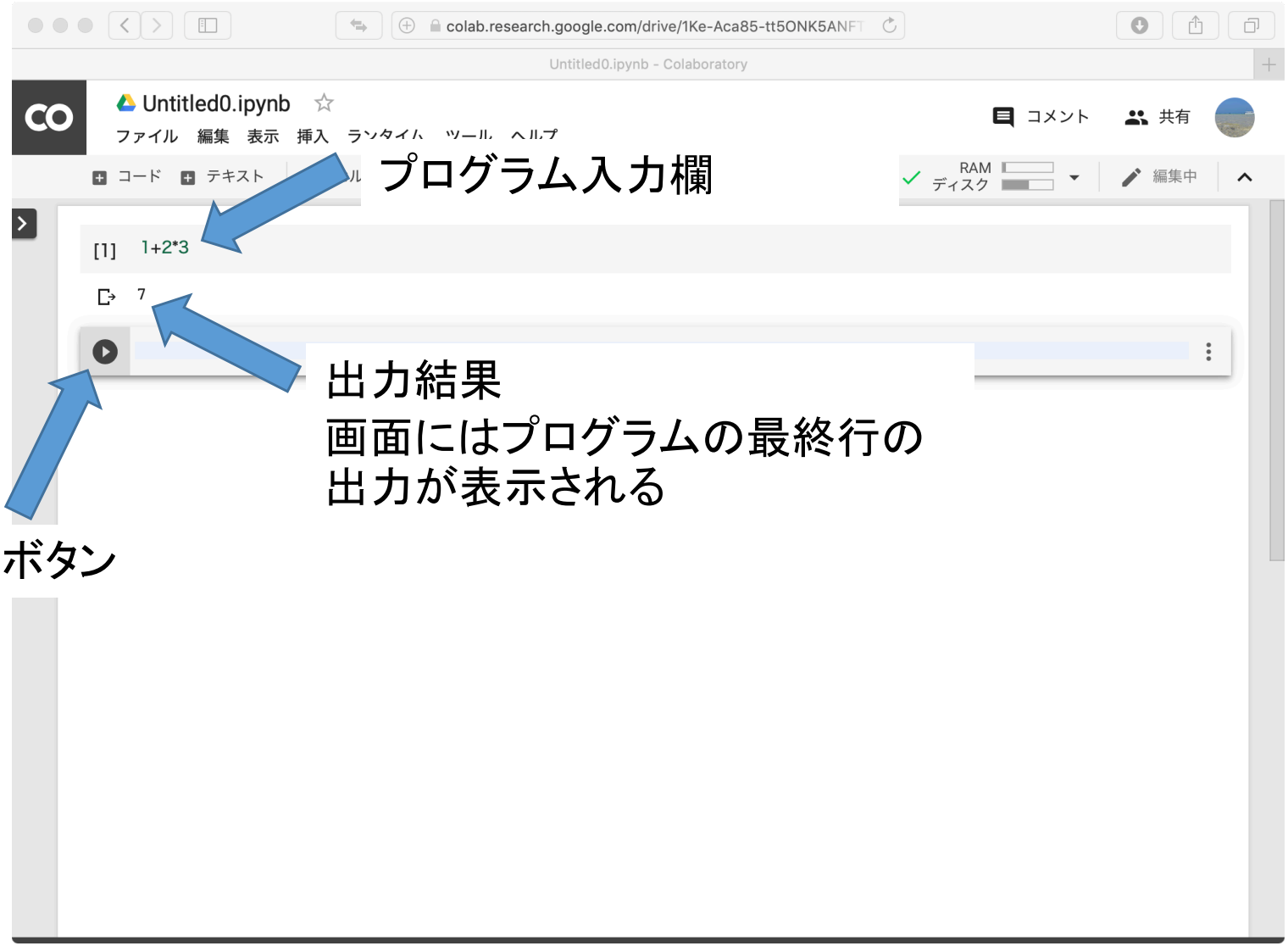
[新規作成] 新しいノートブック

新しいノートブックを選択



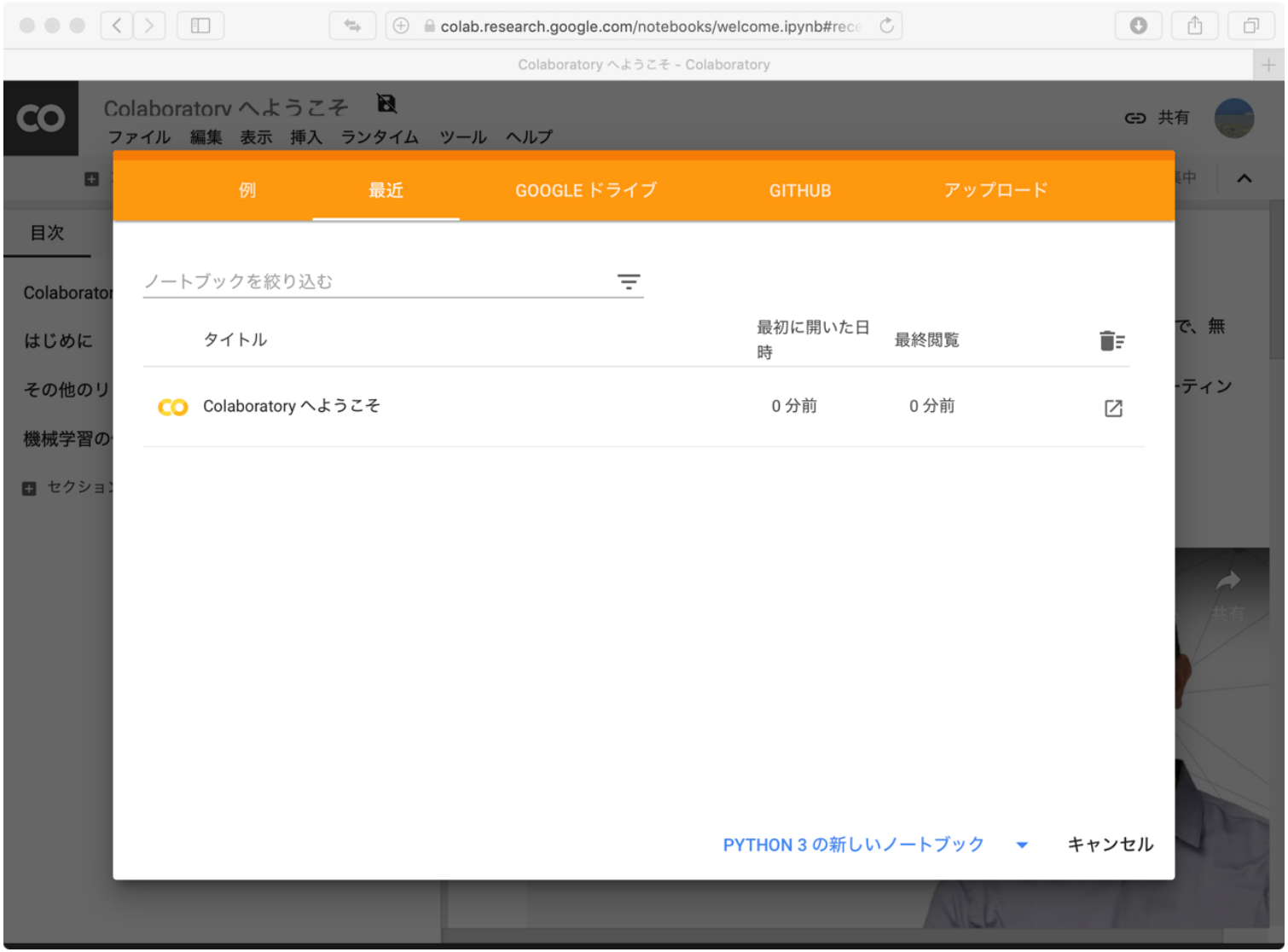
[新規作成] プログラムの実行

プログラムを書いて、実行(　)を押す



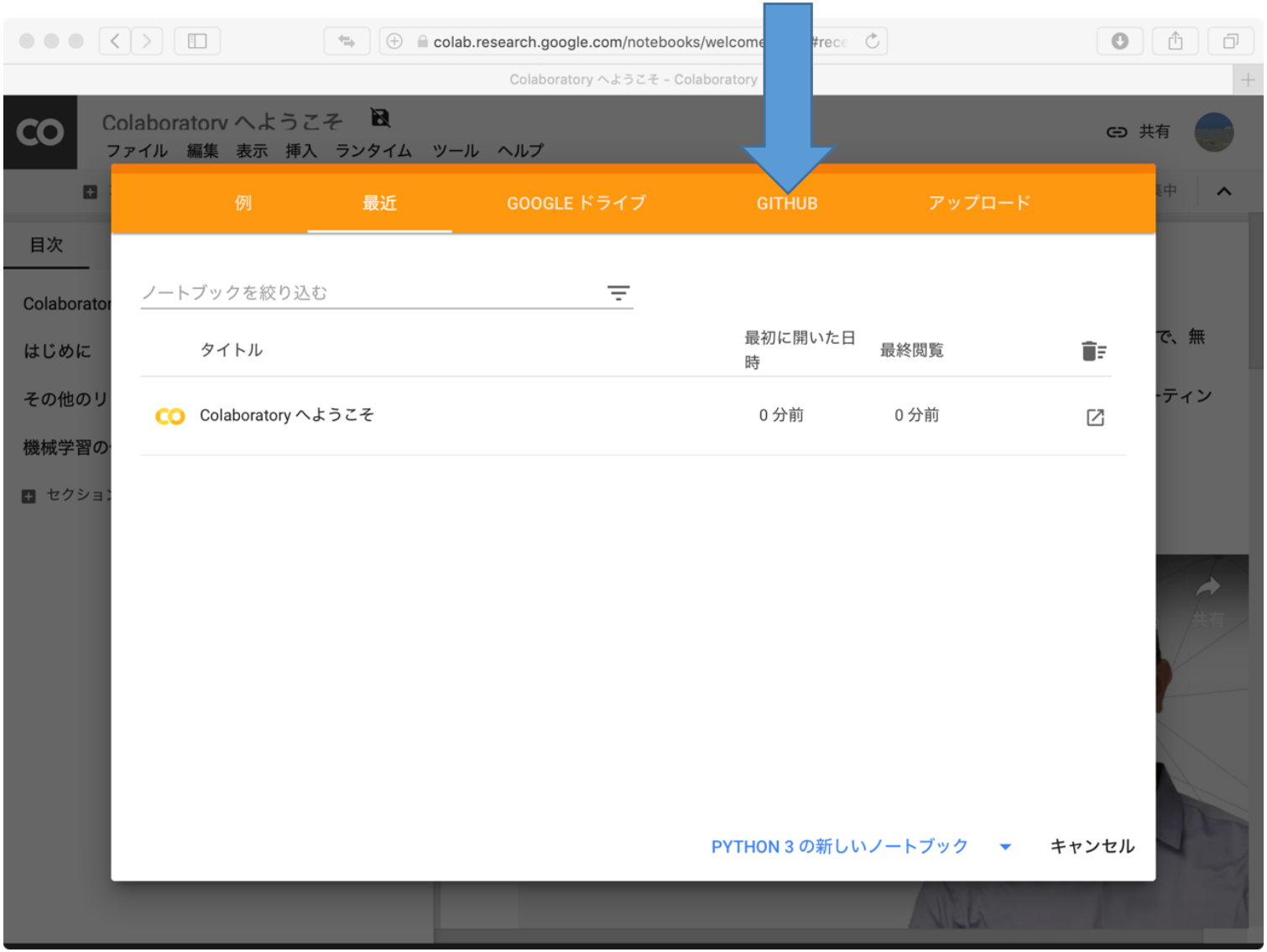
[レシピ実行] Githubから開く

https://colab.research.google.com/ にアクセス



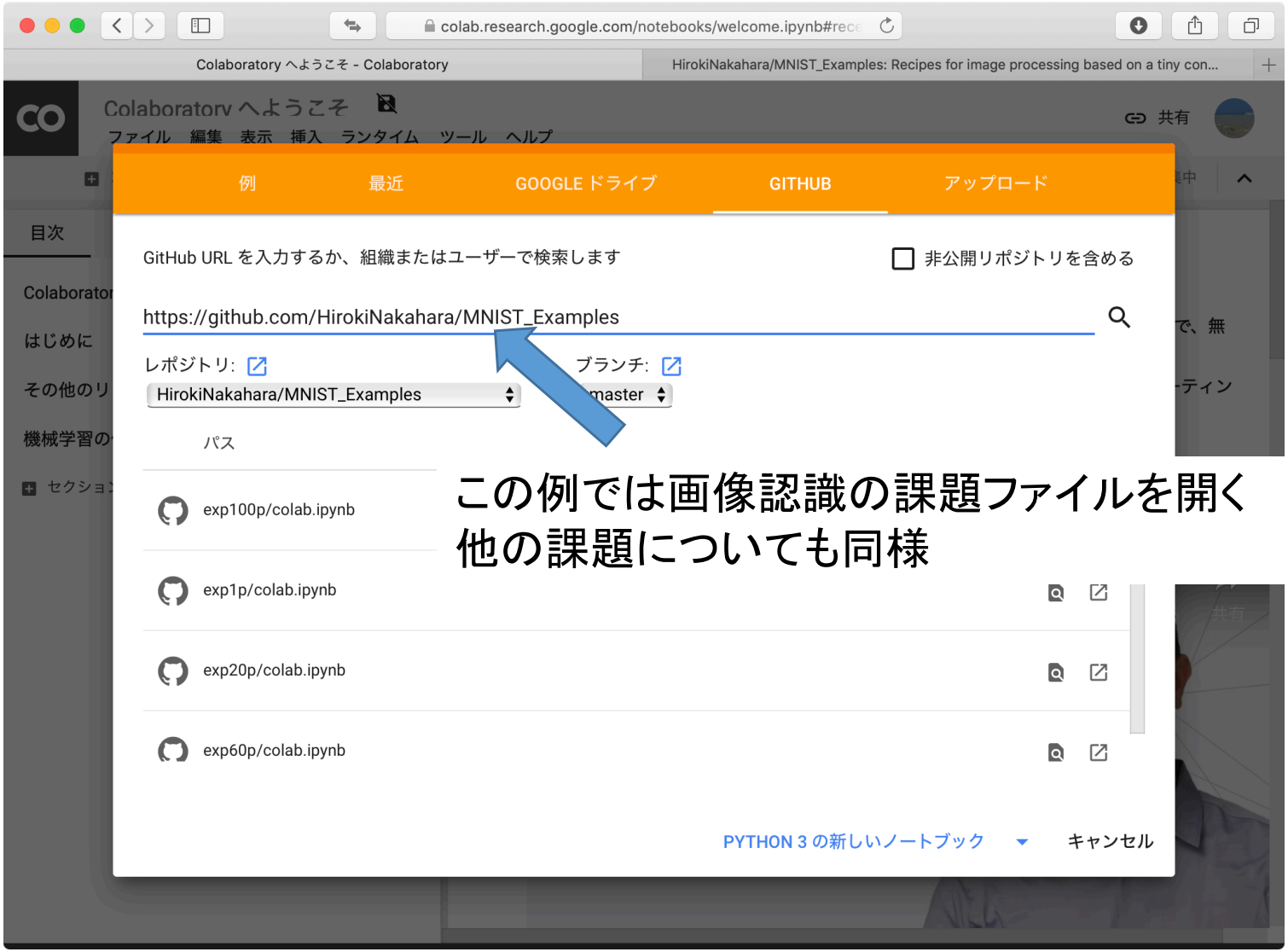
[レシピ実行] Githubから開く

GITHUBを選択



[レシピ実行] Githubから開く

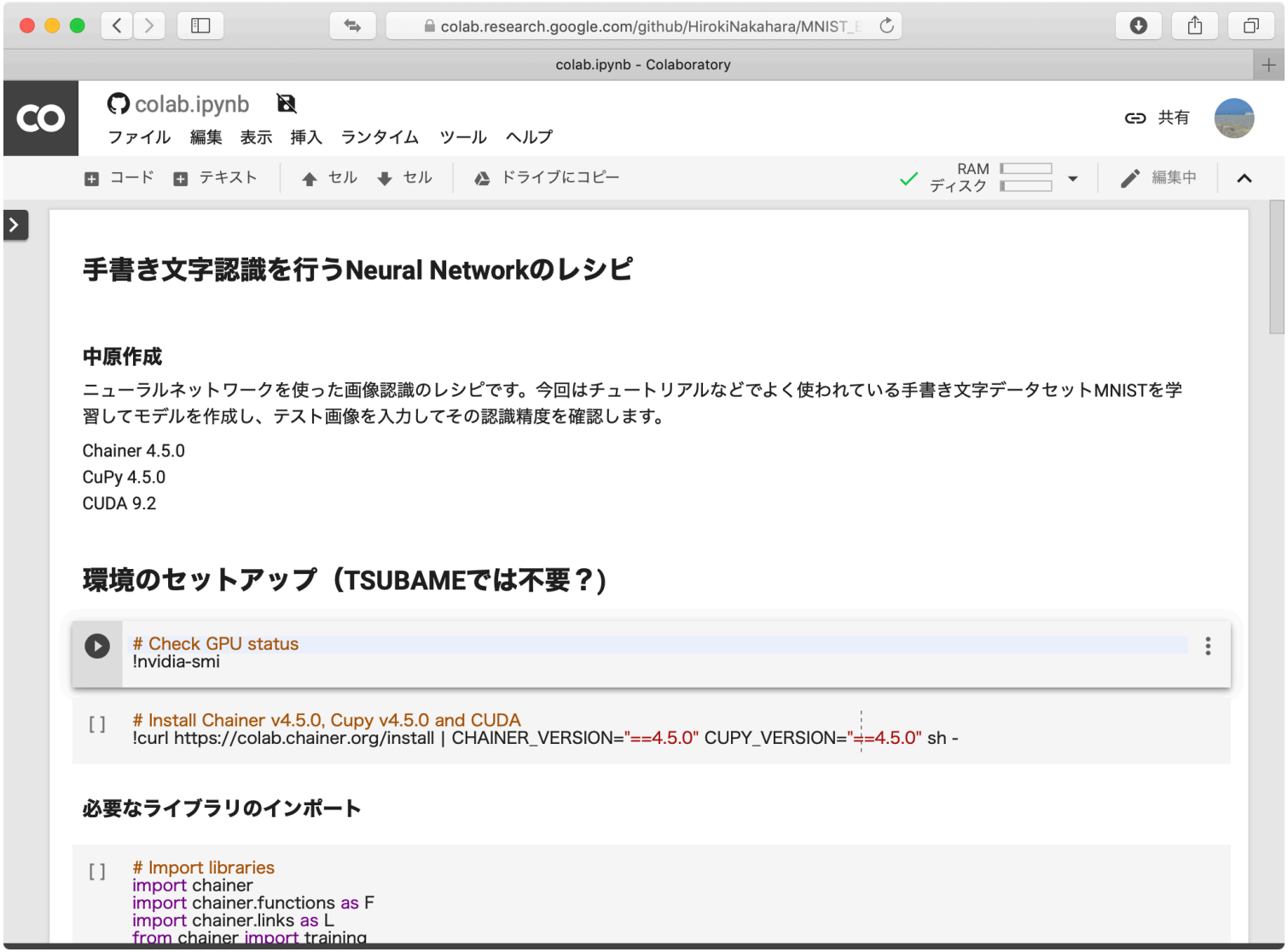
GithubのURLを入力してファイルを選択



この例では画像認識の課題ファイルを開く
他の課題についても同様

[レシピ実行] プログラムの実行

順番に実行(　)を押す

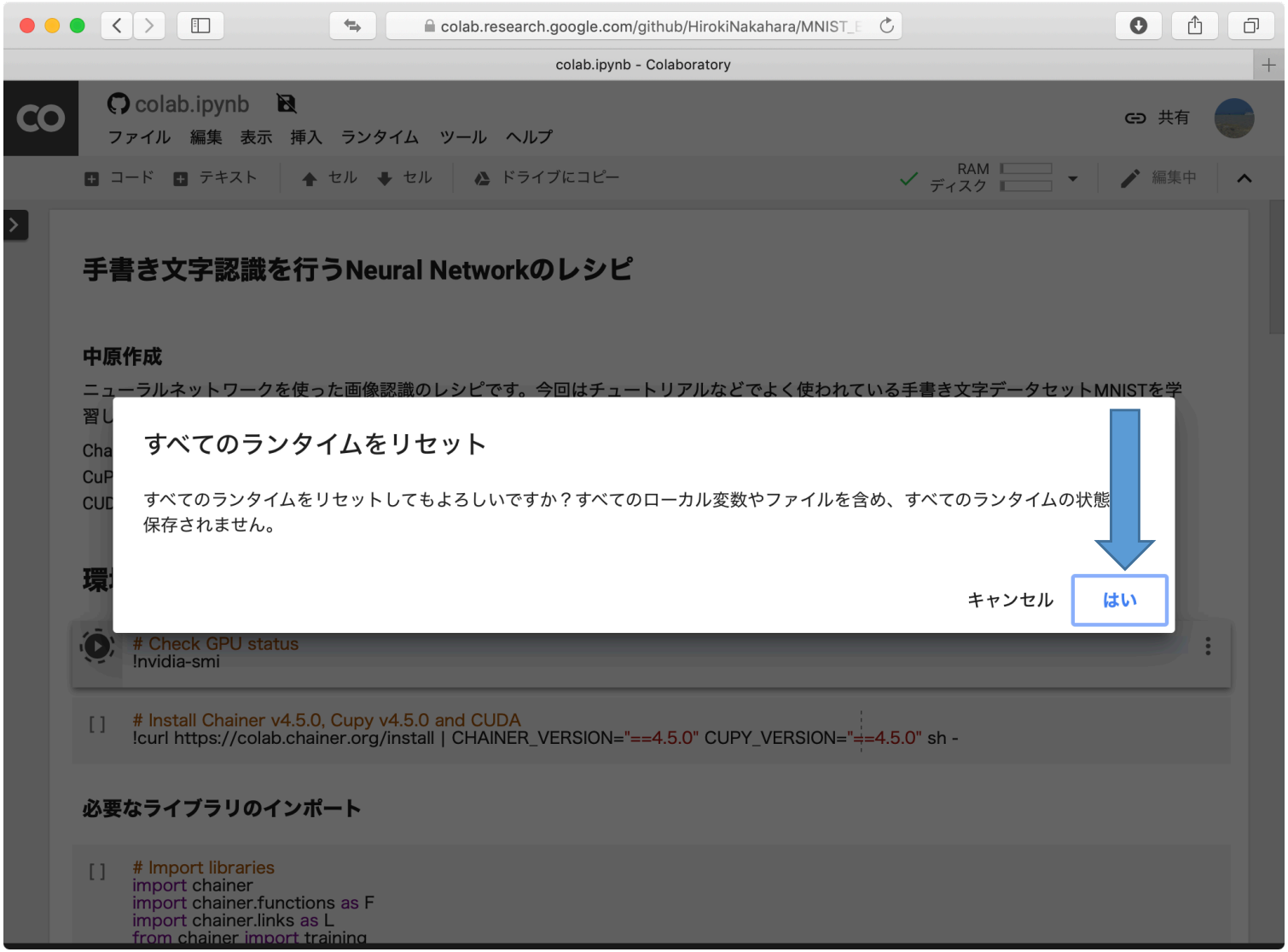


[レシピ実行] プログラムの実行

初回は警告が出るので「このまま実行」を押す

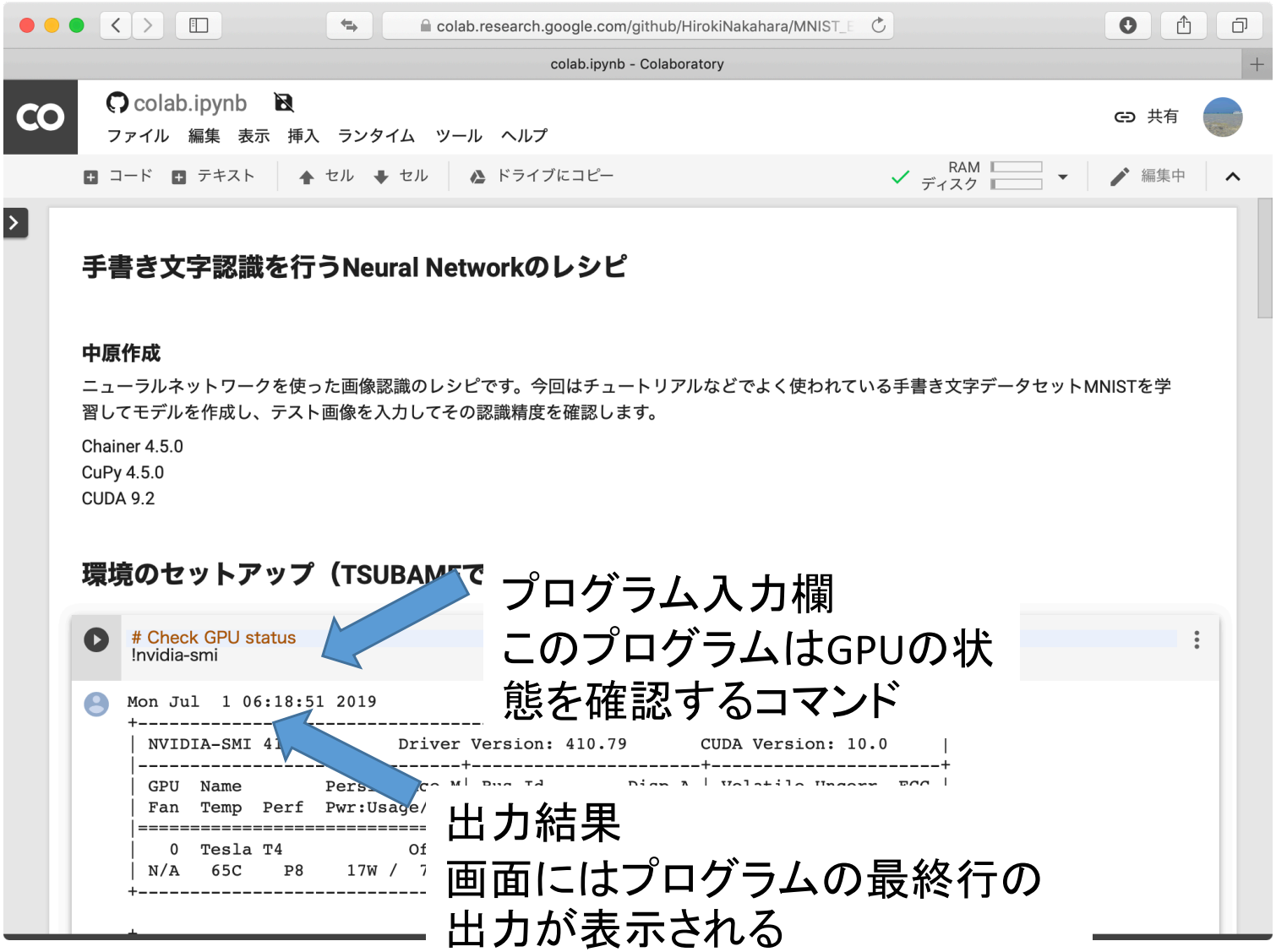
[レシピ実行] プログラムの実行

同様に「はい」を押す



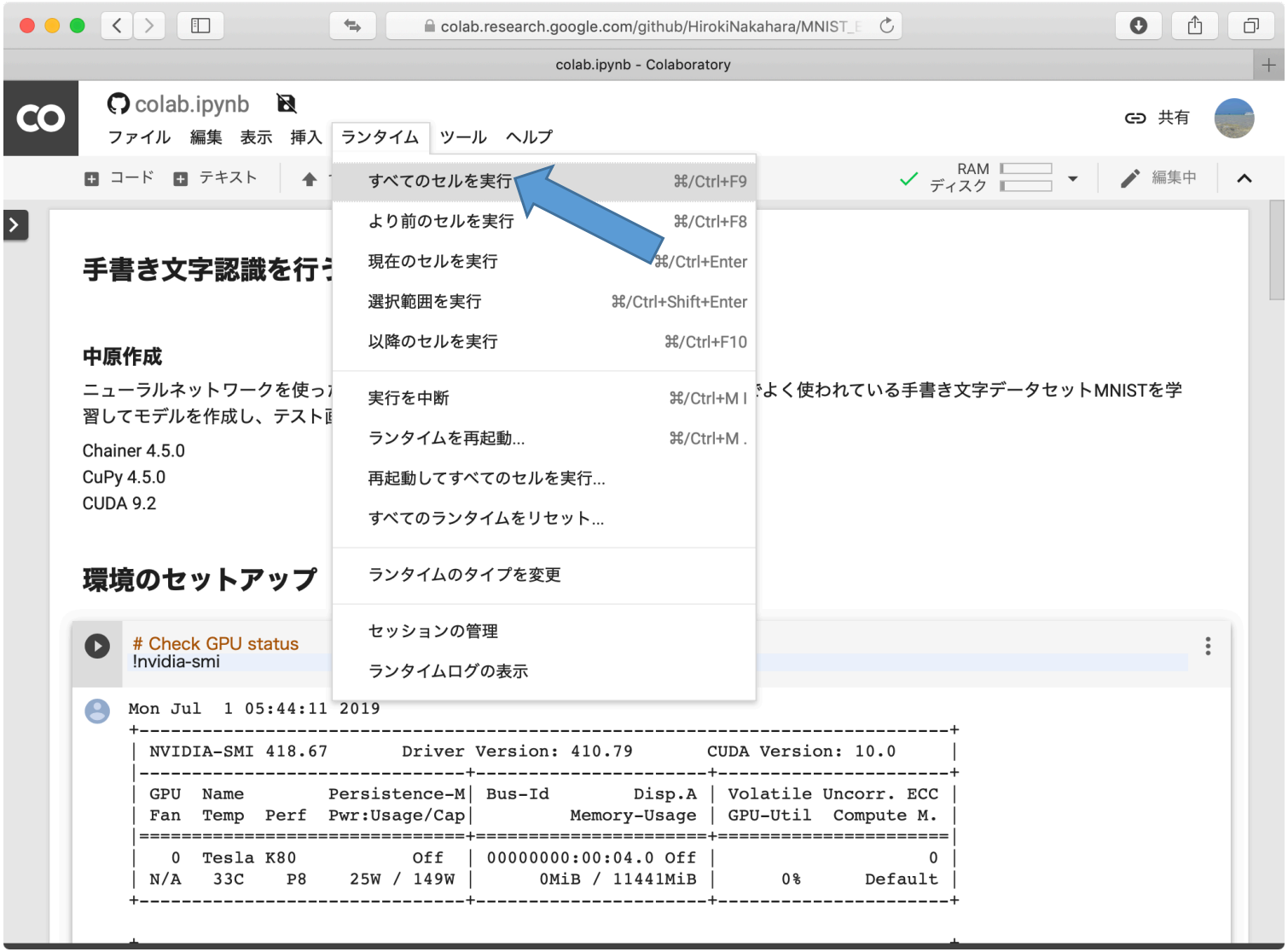
[レシピ実行] プログラムの実行

プログラムの出力が表示される



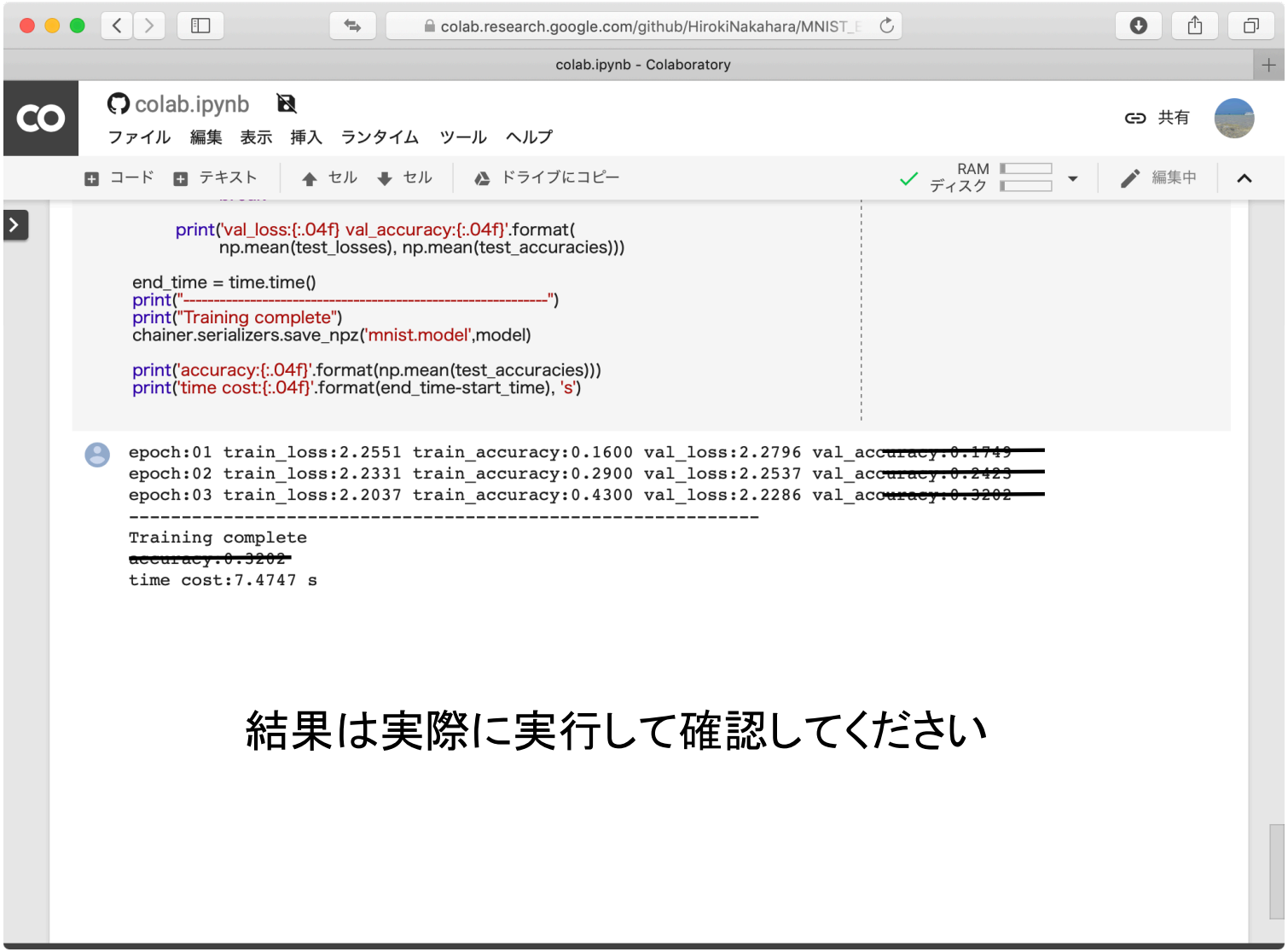
[レシピ実行] プログラムの実行

全てのプログラムを実行することもできる



[レシピ実行] 学習結果の確認

授業プログラムは最後のセルに学習結果が出力



結果は実際に実行して確認してください