

FileEditViewNavigateCodeRefactorRunToolsVCSWindowHelpAssignment-1 - problem_5.py

Assignment-1 > problem_5.py

main

problem_5.py

Project

Assignment-1 C:\Users\tmart\Down

example_file.txt

main.py

problem_1.py

problem_2.py

problem_3.py

problem_4.py

problem_5.py

problem_6.py

External Libraries

Scratches and Consoles

main.py

problem_1.py

problem_2.py

problem_3.py

problem_4.py

problem_5.py

problem_6.py

example_file.txt

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

Precondition: input is a non-empty string

def string_alternative(input_string):

Return a string of every other member of the input string using string splicing

return input_string[::2]

def solution(line_break):

Print line_break and problem number to console

print(line_break + "Problem 5: \n")

Message prompts for user input

message_1 = "Enter a string input: "

message_2 = "User input invalid. Please re-enter a valid string input: "

Grab input from user

user_input = input(message_1)

Ensure input is not empty

while not len(user_input):

user_input = input(message_2)

Output string from a call to user-defined function 'string_alternative' to console

print("Output: " + string_alternative(user_input))

print()

Run

TODO

Problems

Debug

Terminal

Python Packages

Python Console

Event Log

Type here to search

10:40 CRLF UTF-8 4 spaces Python 3.8 (tf)

1:28 PM 6/14/2021

FileEditViewNavigateCodeRefactorRunToolsVCSWindowHelp

Assignment-1 - problem_1.py

Assignment-1 > problem_1.py

main

▶

🐛

🔄

🔍

⚙️

Project

Project

Assignment-1 C:\Users\tmart\Down

example_file.txt

main.py

problem_1.py

problem_2.py

problem_3.py

problem_4.py

problem_5.py

problem_6.py

External Libraries

Scratches and Consoles

main.py × problem_1.py × problem_2.py × problem_3.py × problem_4.py × problem_5.py × problem_6.py × example_file.txt ×

1import random

2

3

4def solution(line_break):

5# Print line_break and problem number to console

6print(line_break + "Problem 1: \n")

7

8# Number of letters to delete, feel free to change

9num_letters_to_delete = 2

10

11# Message prompt for user input

12message = "Enter desired word(s): "

13

14# Grab input from user, instantiating a list in which to store input

15letters_list = list(input(message))

16

17# For as many letters as we wish to delete (Instructions say 2)

18for x in range(num_letters_to_delete):

19# If there is a member of letters_list (this ensures an IndexError doesn't happen for an empty list)

20if len(letters_list):

21# Pop a random index from letters_list

22letters_list.pop(random.randint(0, len(letters_list) - 1))

23

24# Join list to a string, then reverse string, then output to console

25print("Output: " + ''.join(letters_list)[::-1])

26print()

27

Run

TODO

Problems

Debug

Terminal

Python Packages

Python Console

Event Log

27:1 CRLF UTF-8 4 spaces Python 3.8 (tf)

1:27 PM 6/14/2021

FileEditViewNavigateCodeRefactorRunToolsVCSWindowHelp

Assignment-1 - problem_2.py

Assignment-1 > problem_2.py

Project

Project

Assignment-1 C:\Users\tmart\Down

example_file.txt

main.py

problem_1.py

problem_2.py

problem_3.py

problem_4.py

problem_5.py

problem_6.py

External Libraries

Scratches and Consoles

main.py ×

problem_1.py ×

problem_2.py ×

problem_3.py ×

problem_4.py ×

problem_5.py ×

problem_6.py ×

example_file.txt ×

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

```
def solution(line_break):  
    # Print line_break and problem number to console  
    print(line_break + "Problem 2: \n")  
  
    # Message prompts for user input  
    message_1 = "Please enter a real number: "  
    message_2 = "Please enter another real number: "  
    message_3 = "User input was not a real number. Please try again..."  
  
    while True:  
        # Grab input from user  
        x = input(message_1)  
        try:  
            # Attempt to cast to int  
            x = int(x)  
            # If no error, break out of while loop  
            break  
        except ValueError:  
            # If we raise an error while attempting to cast into int  
            try:  
                # Attempt to cast to float  
                x = float(x)  
                # If no error, break out of while loop  
                break  
            except ValueError:  
                # If both casts (to int and to float) raise an error, then the input is not valid  
                print(message_3)  
  
    while True:  
        solution()
```

Run

TODO

Problems

Debug

Terminal

Python Packages

Python Console

47:48

CRLF

UTF-8

4 spaces

Python 3.8 (tf)

1:28 PM

6/14/2021

FileEditViewNavigateCodeRefactorRunToolsVCSWindowHelp

Assignment-1 - problem_2.py

Assignment-1 > problem_2.py

main

▶

🐛

🔄

🔍

⚙️

Project

Project

Assignment-1 C:\Users\tmart\Down

example_file.txt

main.py

problem_1.py

problem_2.py

problem_3.py

problem_4.py

problem_5.py

problem_6.py

External Libraries

Scratches and Consoles

main.py × problem_1.py × problem_2.py × problem_3.py × problem_4.py × problem_5.py × problem_6.py × example_file.txt ×

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

If both casts (to int and to float) raise an error, then the input is not valid

print(message_3)

while True:

Grab input from user

y = input(message_2)

try:

Attempt to cast to int

y = int(y)

If no error, break out of while loop

break

except ValueError:

If we raise an error while attempting to cast into int

try:

Attempt to cast to float

y = float(y)

If no error, break out of while loop

break

except ValueError:

If both casts (to int and to float) raise an error, then the input is not valid

print(message_3)

Produces output to console for addition, multiplication, raising by power, and floor division

print("Output: \n")

print(f"1. {x} added to {y} = {x + y}")

print(f"2. {x} multiplied by {y} = {x * y}")

print(f"3. {x} to the power of {y} = {x ** y}")

print(f"4. The floor of {x} divided by {y} = {x // y}")

print()

solution() > while True > except ValueError > except ValueError

Run

TODO

Problems

Debug

Terminal

Python Packages

Python Console

47:48 CRLF UTF-8 4 spaces Python 3.8 (tf)

1:28 PM 6/14/2021

FileEditViewNavigateCodeRefactorRunToolsVCSWindowHelp

Assignment-1 - problem_3.py

Assignment-1 > problem_3.py

main

▶

🐛

🔄

🔍

⚙️

Project

Project

Assignment-1 C:\Users\tmart\Down

example_file.txt

main.py

problem_1.py

problem_2.py

problem_3.py

problem_4.py

problem_5.py

problem_6.py

External Libraries

Scratches and Consoles

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

```
def solution(line_break):  
    # Print line_break and problem number to console  
    print(line_break + "Problem 3: \n")  
  
    # Declare and define dictionary of words to search for and their replacements  
    word_dict = {"python": "pythons", "Python": "Pythons"}  
  
    # Message prompt for user input  
    message = "Enter a sentence: "  
  
    # Grab input from user  
    output = input(message)  
  
    # For each entry in the dictionary of search words  
    for word, replacement in word_dict.items():  
        # Use str.replace() to find and replace all instances of keywords  
        output = output.replace(word, replacement)  
  
    # Output result to console  
    print("Output: " + output)  
    print()
```

Run

TODO

Problems

Debug

Terminal

Python Packages

Python Console

Event Log

24:1 CRLF UTF-8 4 spaces Python 3.8 (tf)

1:28 PM 6/14/2021

FileEditViewNavigateCodeRefactorRunToolsVCSWindowHelpAssignment-1 - problem_4.py

Assignment-1 > problem_4.py

main

main.py × problem_1.py × problem_2.py × problem_3.py × problem_4.py × problem_5.py × problem_6.py × example_file.txt ×

Project

Project

Assignment-1 C:\Users\tmart\Down

example_file.txt

main.py

problem_1.py

problem_2.py

problem_3.py

problem_4.py

problem_5.py

problem_6.py

External Libraries

Scratches and Consoles

30

Grab input from user, directly casting it to float

31

NOTE: by using range(len(raw_list), we will initially start at 0,

32

but if we error on an input, we only have to redo that input, and continue from there

33

for index in range(len(raw_list), num_students):

34

raw_list.append(float(input(message_2.format(index + 1))))

35

If no error, break out of while loop

36

break

37

except ValueError:

38

If cast to float raises an error, then the last input is not valid

39

print(message_4)

40

41

Declare list for looped calculation

42

looped_list = []

43

For each weight in the raw_list

44

for weight in raw_list:

45

Append each weight after converting lbs to kg and rounding to 2 decimal places

46

looped_list.append(round(weight * lbs_to_kg, 2))

47

48

Output looped list to console

49

print("\nLooped list: ")

50

print(*looped_list, sep=", ")

51

print()

52

53

Declare and define list for comprehension calculation

54

For each weight in raw_list, convert from lbs to kg, then round to 2 decimal places

55

comprehension_list = [round(weight * lbs_to_kg, 2) for weight in raw_list]

56

57

Output comprehension list to console

58

print("Comprehension list: ")

59

print(*comprehension_list, sep=", ")

60

print()

61

Run

TODO

Problems

Debug

Terminal

Python Packages

Python Console

Event Log

Type here to search

45:69 CRLF UTF-8 4 spaces Python 3.8 (tf)

1:28 PM 6/14/2021

FileEditViewNavigateCodeRefactorRunToolsVCSWindowHelp

Assignment-1 - problem_6.py

Assignment-1 > problem_6.py

main

▶

🐛

🔄

🔍

⚙️

Project

Project

Assignment-1 C:\Users\tmart\Down

example_file.txt

main.py

problem_1.py

problem_2.py

problem_3.py

problem_4.py

problem_5.py

problem_6.py

External Libraries

Scratches and Consoles

main.py × problem_1.py × problem_2.py × problem_3.py × problem_4.py × problem_5.py × problem_6.py × example_file.txt ×

1import re

2

3

4def solution(line_break):

5# Print line_break and problem number to console

6print(line_break + "Problem 6: \n")

7

8# I decided to put a pause for the beginning of problem 6,

9# since it doesn't require user input, it's easy for the console to 'get ahead' of the user

10message = "Problem 6 does not ask for user input, please press any key to continue..."

11input(message)

12

13# Name (and path, if required) of file to process

14filename = "example_file.txt"

15# Declare dictionary to store words counted (key) and their counts (value)

16word_dict = {}

17

18# Open file to read with auto-close method

19with open(filename, 'r') as file:

20for line in file:

21# Discard all non-alphanumeric and non-whitespace characters

22scrubbed_line = re.sub(r"^[^w\s]", "", line)

23# Discard all new lines

24scrubbed_line = re.sub(r"\n", "", scrubbed_line)

25# Create list from split string

26words_in_line = scrubbed_line.split(" ")

27

28for word in words_in_line:

29# Modify for consistent key capitalization

30cap_word = word.lower().capitalize()

31# If the word is in the dictionary

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

156

157

158

159

160

161

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

286

287

288

289

290

291

292

293

294

295

296

297

298

299

300

301

302

303

304

305

306

307

308

309

310

311

312

313

314

315

316

317

318

319

320

321

322

323

324

325

326

327

328

329

330

331

332

333

334

335

336

337

338

339

340

341

342

343

344

345

346

347

348

349

350

351

352

353

354

355

356

357

358

359

360

361

362

363

364

365

366

367

368

369

370

371

372

373

374

375

376

377

378

379

380

381

382

383

384

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

402

403

404

405

406

407

408

409

410

411

412

413

414

415

416

417

418

419

420

421

422

423

424

425

426

427

428

429

430

431

432

433

434

435

436

437

438

439

440

441

442

443

444

445

446

447

448

449

450

451

452

453

454

455

456

457

458

459

460

461

462

463

464

465

466

467

468

469

470

471

472

473

474

475

476

477

478

479

480

481

482

483

484

485

486

487

488

489

490

491

492

493

494

495

496

497

498

499

500

501

502

503

504

505

506

507

508

509

510

511

512

513

514

515

516

517

518

519

520

521

522

523

524

525

526

527

528

529

530

531

532

533

534

535

536

537

538

539

540

541

542

543

544

545

546

547

548

549

550

551

552

553

554

555

556

557

558

559

560

561

562

563

564

565

566

567

568

569

570

571

572

573

574

575

576

577

578

579

580

581

582

583

584

585

586

587

588

589

590

591

592

593

594

595

596

597

598

599

600

601

602

603

604

605

606

607

608

609

610

611

612

613

614

615

616

617

618

619

620

621

622

623

624

625

626

627

628

629

630

631

632

633

634

635

636

637

638

639

640

641

642

643

644

645

646

647

648

649

650

651

652

653

654

655

656

657

658

659

660

661

662

663

664

665

666

667

668

669

670

671

672

673

674

675

676

677

678

679

680

681

682

683

684

685

686

687

688

689

690

691

692

693

694

695

696

697

698

699

700

701

702

703

704

705

706

707

708

709

710

711

712

713

714

715

716

717

718

719

720

721

722

723

724

725

726

727

728

729

730

731

732

733

734

735

736

737

738

739

740

741

742

743

744

745

746

747

748

749

750

751

752

753

754

755

756

757

758

759

760

761

762

763

764

765

766

767

768

769

770

771

772

773

774

775

776

777

778

779

780

781

782

783

784

785

786

787

788

789

790

791

792

793

794

795

796

797

798

799

800

801

802

803

804

805

806

807

808

809

810

811

812

813

814

815

816

817

818

819

820

821

822

823

824

825

826

827

828

829

830

831

832

833

834

835

836

837

838

839

840

841

842

843

844

845

846

847

848

849

850

851

852

853

854

855

856

857

858

859

860

861

862

863

864

865

866

867

868

869

870

871

872

873

874

875

876

877

878

879

880

881

882

883

884

885

886

887

888

889

890

891

892

893

894

895

896

897

898

899

900

901

902

903

904

905

906

907

908

909

910

911

912

913

914

915

916

917

918

919

920

921

922

923

924

925

926

927

928

929

930

931

932

933

934

935

936

937

938

939

940

941

942

943

944

945

946

947

948

949

950

951

952

953

954

955

956

957

958

959

960

961

962

963

964

965

966

967

968

969

970

971

972

973

974

975

976

977

978

979

980

981

982

983

984

985

986

987

988

989

990

991

992

993

994

995

996

997

998

999

1000

2000

2001

2002

2003

2004

2005

2006

2007

2008

2009

2010

2011

2012

2013

2014

2015

2016

2017

2018

2019

2020

2021

2022

2023

2024

2025

2026

2027

2028

2029

2030

2031

2032

2033

2034

2035

2036

2037

2038

2039

2040

2041

2042

2043

2044

2045

2046

2047

2048

2049

2050

2051

2052

2053

2054

2055

2056

2057

2058

2059

2060

2061

2062

2063

2064

2065

2066

2067

2068

2069

2070

2071

2072

2073

2074

2075

2076

2077

2078

2079

2080

2081

2082

2083

2084

2085

2086

2087

2088

2089

2090

2091

2092

2093

2094

2095

2096

2097

2098

2099

2100

2101

2102

2103

2104

2105

2106

2107

2108

2109

2110

2111

2112

2113

2114

2115

2116

2117

2118

2119

2120

2121

2122

2123

2124

2125

2126

2127

2128

2129

2130

2131

2132

2133

2134

2135

2136

2137

2138

2139

2140

2141

2142

2143

2144

2145

2146

2147

2148

2149

2150

2151

2152

2153

2154

2155

2156

2157

2158

2159

2160

2161

2162

2163

2164

2165

2166

2167

2168

2169

2170

2171

2172

2173

2174

2175

2176

2177

2178

2179

2180

2181

2182

2183

2184

2185

2186

2187

2188

2189

2190

2191

2192

2193

2194

2195

2196

2197

2198

2199

2200

2201

2202

2203

2204

2205

2206

2207

2208

2209

2210

2211

2212

2213

2214

2215

2216

2217

2218

2219

2220

2221

2222

2223

2224

2225

2226

2227

2228

2229

2230

2231

2232

2233

2234

2235

2236

2237

2238

2239

2240

2241

2242

2243

2244

2245

2246

2247

2248

2249

2250

2251

2252

2253

2254

2255

2256

2257

2258

2259

2260

2261

2262

2263

2264

2265

2266

2267

2268

2269

2270

2271

2272

2273

2274

2275

2276

2277

2278

2279

2280

2281

2282

2283

2284

2285

2286

2287

2288

2289

2290

2291

2292

2293

2294

2295

2296

2297

2298

2299

2300

2301

2302

2303

2304

2305

2306

2307

2308

2309

2310

2311

2312

2313

2314

2315

2316

2317

2318

2319

2320

2321

2322

2323

2324

2325

2326

2327

2328

2329

2330

2331

2332

2333

2334

2335

2336

2337

2338

2339

2340

2341

2342

2343

2344

2345

2346

2347

2348

2349

2350

2351

2352

2353

2354

2355

2356

2357

2358

2359

2360

2361

2362

2363

2364

2365

2366

2367

2368

2369

2370

2371

2372

2373

2374

2375

2376

2377

2378

2379

2380

2381

2382

2383

2384

2385

2386

2387

2388

2389

2390

2391

2392

2393

2394

2395

2396

2397

2398

2399

2400

2401

2402

2403

2404

2405

2406

2407

2408

2409

2410

2411

2412

2413

2414

2415

2416

2417

2418

2419

2420

2421

2422

2423

2424

2425

2426

2427

2428

2429

2430

2431

2432

2433

2434

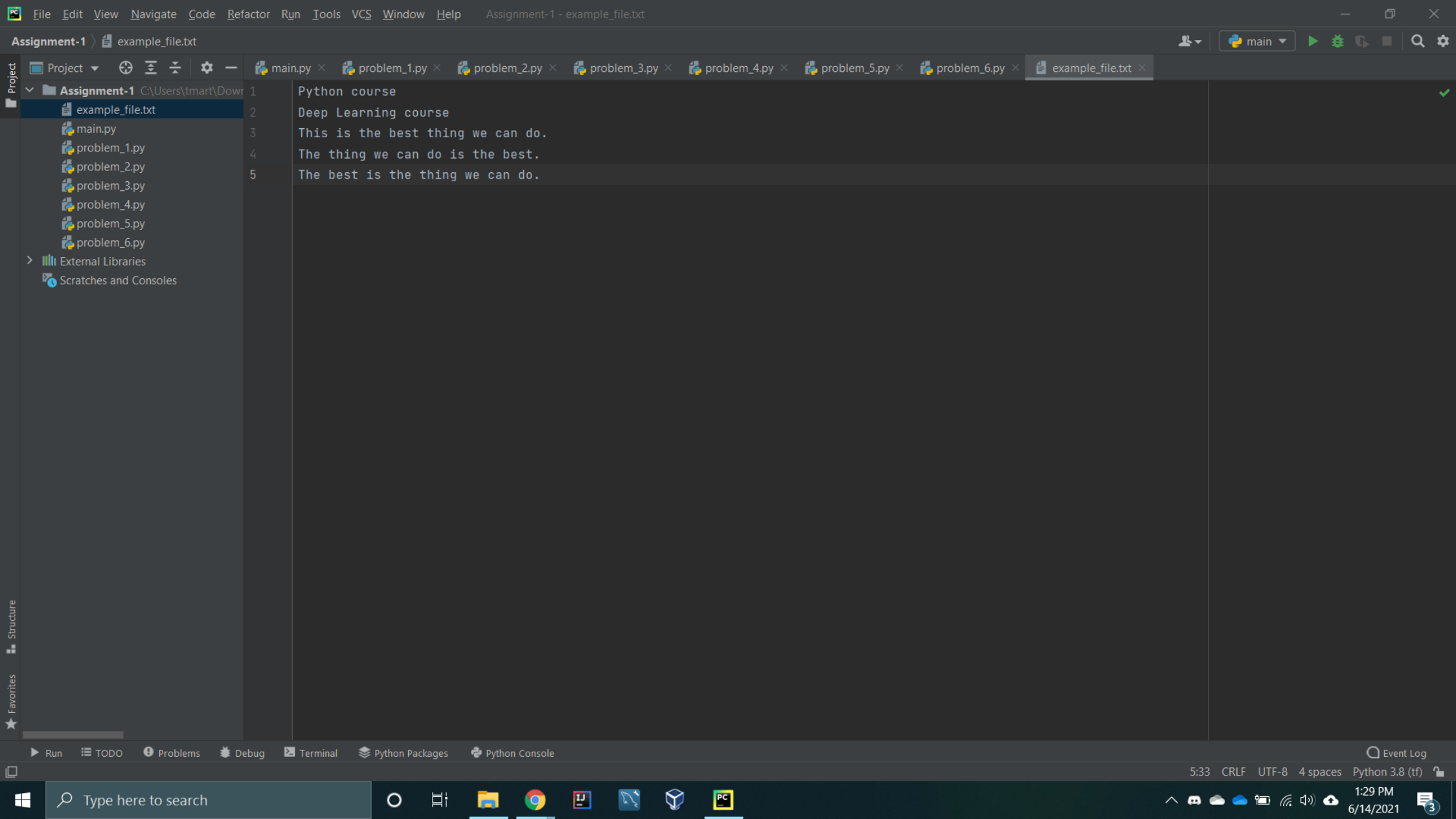
2435

2436

2437

2438

- ★ Favorites
- ■ ■ Structure



FileEditViewNavigateCodeRefactorRunToolsGitWindowHelp

assignment-1-tmartinweb - main.py

assignment-1-tmartinwebAssignment-1main.py

Project

assignment-1-tmartinwebC:\Users\tmart\Downloads\UMKC\

Run:main

C:\Users\tmart\anaconda3\envs\tf\python.exe C:/Users/tmart/Downloads/UMKC/CS-490-DeepLearning/assignment-1-tmartinweb/Assignment-1/main.py

CommitPull RequestStructureFavorites

Solutions for Assignment 1:

Problem 1:

Enter desired word(s): python

Output: nohy

Problem 2:

Please enter a real number: 18

Please enter another real number: 4.5

Output:

1. 18 added to 4.5 = 22.5

2. 18 multiplied by 4.5 = 81.0

3. 18 to the power of 4.5 = 445375.4487710341

4. The floor of 18 divided by 4.5 = 4.0

Problem 3:

Enter a sentence: I love Python, but don't want to meet a python.

Output: I love Pythons, but don't want to meet a pythons.

Problem 4:

GitRunTODOProblemsTerminalPython PackagesPython Console

Pushed 1 commit to origin/master (3 minutes ago)

Updating skeletons...67:1 Python 3.8 master

Type here to search

Taskbar icons: File Explorer, Chrome, VS Code, etc.

System tray: 1:56 PM, 6/14/2021, Event Log, 3 notifications

FileEditViewNavigateCodeRefactorRunToolsGitWindowHelpassignment-1-tmartinweb - main.py

assignment-1-tmartinwebAssignment-1main.py

Projectassignment-1-tmartinwebC:\Users\smart\Downloads\UMKC\VC

Run: main

ProjectCommitPull RequestsStructureFavorites

Project

assignment-1-tmartinweb

Run: main

Enter the #4 student's weight (in lbs.): 150

Looped list:

45.36, 34.02, 22.68, 68.04

Comprehension list:

45.36, 34.02, 22.68, 68.04

#####

Problem 5:

Enter a string input: Good evening

Output: Go vnn

#####

Problem 6:

Problem 6 does not ask for user input, please press any key to continue...

Word_Count:

{'Python': 1, 'Course': 2, 'Deep': 1, 'Learning': 1, 'This': 1, 'Is': 3, 'The': 5, 'Best': 3, 'Thing': 3, 'We': 3, 'Can': 3, 'Do': 3}

Word count appended to file.

#####

--End of Program --

#####

Process finished with exit code 0

GitRunTODOProblemsTerminalPython PackagesPython Console

Pushed 1 commit to origin/master (4 minutes ago)

Updating skeletons...67:1 Python 3.8 master

Event Log

Type here to search

Taskbar icons

System tray

