

Diff:

Differences between given skeleton and solution

In order to make the sample solution easier to understand, the differences between it and the given skeleton source code were highlighted with the help of the program `diff`.

Legend:

- Gray: unchanged text (only excerpts).
- Green: new lines
- Yellow: changed lines
- Red: deleted lines

Note: Files not listed have not been changed.

This document was created with the help of [diff2html](#) erstellt.

Nur in ../course11-gui-part1/exercise/solution/: parametervalues.ini.

Gemeinsame Unterverzeichnisse: ../course11-gui-part1/exercise/code/__pycache__ und ../course11-gui-part1/exercise/solution/__pycache__.

diff -u ../course11-gui-part1/exercise/code/simgui.py ../course11-gui-part1/exercise/solution/simgui.py

../course11-gui-part1/exercise/code/simgui.py	../course11-gui-part1/exercise/solution/simgui.py
17	17
18 import matplotlib.pyplot as plt	18 import matplotlib.pyplot as plt
19	19
	20 import configparser
	21
20	22
21 # QApplication instance is always needed (just accept sys.argv)	23 # QApplication instance is always needed (just accept sys.argv)
22 app = QtWidgets.QApplication(sys.argv)	24 app = QtWidgets.QApplication(sys.argv)
23	25
24	26
25	27
26	28
27	29
28	30
29	31
30	32
31	33
32	34
33	35
34	36
35	37
36	38
37	39
38	40
39	41
40	42
41	43
42	44
43	45
44	46
45	47
46	48
47	49
48	50
49	51
50	52
51	53
52	54
53	55
54	56
55	57
56	58
57	59
58	60
59	61
60	62
61	63
62	64
63	65
64	66
65	67
66	68
67	69
68	70
69	71
70	72
71	73
72	74
73	75
74	76
75	77
76	78
77	79
78	80
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	
7	

```

56
57
58 # task 11.1.5
59 # limit input characters (only float numbers should be allowed)
60 # ...
61
62 # task 11.1.6
63 # set alignment
64 # ....
65
66
67 # optional: set focus to the exit button
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85 # Create ConfigParser and pass data
86 c = configparser.ConfigParser()
87 c.set("XXX", "XXX", mass1_edit.text())
88 c.XXX
89
90
91 # write config file
92 with open(filename, 'w') as fid:
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116 print("No configuration file loaded")
117
118 # pass values to the according QLineEdit instances
119 # mass1_edit.setText(...).
120 .
121
122
123
124 def simulate():
125
126
127
128
129
130
131
132 # fetch values from the gui
133 m1 = float(mass1_edit.text())
134 # ...

```

```

81
82
83 # task 11.1.5
84 # limit input characters (only float numbers should be allowed)
85 length_edit.setValidator(QtGui.QDoubleValidator(length_edit)).
86 mass1_edit.setValidator(QtGui.QDoubleValidator(mass1_edit))
87 mass2_edit.setValidator(QtGui.QDoubleValidator(mass2_edit))
88 step_size_edit.setValidator(QtGui.QDoubleValidator(step_size_edit))
89 duration_edit.setValidator(QtGui.QDoubleValidator(duration_edit))
90
91
92 # task 11.1.6
93 # set alignment
94 length_edit.setAlignment(QtCore.Qt.AlignRight).
95 mass1_edit.setAlignment(QtCore.Qt.AlignRight)
96 mass2_edit.setAlignment(QtCore.Qt.AlignRight)
97 step_size_edit.setAlignment(QtCore.Qt.AlignRight)
98 duration_edit.setAlignment(QtCore.Qt.AlignRight)
99
100
101 # optional: set focus to the exit button
102
103
104
105
106
107
108
109 # Create ConfigParser and pass data
120 c = configparser.ConfigParser()
121
122
123
124
125
126
127
128
129
130
131 # write config file
132 with open(filename, 'w') as fid:
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156 print("No configuration file loaded")
157
158 # pass values to the according QLineEdit instances
159 mass1_edit.setText(c.get('Parameter', 'm1')).
160 mass2_edit.setText(c.get('Parameter', 'm2')).
161 duration_edit.setText(c.get('Parameter', 'l'))
162
163
164
165
166
167 def simulate():
168
169
170
171
172
173
174
175 # fetch values from the gui
176 m1 = float(mass1_edit.text())
177 m2 = float(mass2_edit.text()).
178 l = float(duration_edit.text())

```

	179 dx = float(step_size_edit.text())
	180 t_end = float(duration_edit.text())
	181
	182 # alternatively:
	183 # m1 = mass1_edit.text().toDouble()[0] # returns tuple like (value, OK)
135	184
136 # create time array	185 # create time array
137 # t = ...	186 t = arange(0, t_end, dx).
138	187
139 # execute simulation (todo: use solve_ivp here) see task description.	188 # execute simulation (todo: use solve_ivp here).
140 # res ...	189 res = odeint(rhs, [0, 0.3, 0, 0], t, args=(m1, m2, l)).
141	190
142 # Plot the results	191 # Plot the results
143 # Here we have to do some trickery: we create a new dialog on which	192 # Here we have to do some trickery: we create a new dialog on which
:	:
151	200
152 # result for the trolley	201 # result for the trolley
153 ax1 = fig.add_subplot(2, 1, 1)	202 ax1 = fig.add_subplot(2, 1, 1)
154 # ...	203 ax1.plot(t, res[:, 0], label='x').
	204 ax1.plot(t, res[:, 2], label='dx')
155	205
	206 ax1.grid(True)
	207 ax1.legend()
	208 ax1.set_ylabel('trolley')
156	209
157 # result for the load	210 # result for the load
158 # ax2 = ...	211 ax2 = fig.add_subplot(2, 1, 2).
	212 ax2.plot(t, res[:, 1], label=r"\$\varphi\$")
	213 ax2.plot(t, res[:, 3], label=r"\$\dot{\varphi}\$")
	214
	215 ax2.grid(True)
	216 ax2.legend()
	217 ax2.set_xlabel('time [s]')
	218 ax2.set_ylabel('load')
159	219
160 # Here now the dialog is displayed and no longer the show function of	220 # Here now the dialog is displayed and no longer the show function of
161 # matplotlib is called	221 # matplotlib is called
162 plot_dialog.show().	222 plotDialog.show().
163	223
164	224
165 # task 11.1.7	225 # task 11.1.7
166 # connect button	226 # connect button
167 # simulation_button.clicked.XXX.	227 simulation_button.clicked.connect(simulate).
168	228
169 # task 11.2.1	229 # task 11.2.1
170 # ...	230 .
	231 open_button = QtWidgets.QPushButton('Open', dialog)
	232 save_button = QtWidgets.QPushButton('Save', dialog)
	233
	234 open_button.clicked.connect(openFile)
	235 save_button.clicked.connect(saveFile)
	236
	237 layout.addWidget(open_button, 7, 1, QtCore.Qt.AlignRight)
	238 layout.addWidget(save_button, 8, 1, QtCore.Qt.AlignRight)
	239
171	240
172 #-----	241 #-----
173	242