

```

1 from os.path import exists
2 from json import dump, load
3
4
5 def main() -> None:
6     splash_screen()
7     while True:
8         selection = check_option(1, 11, "Select an action: ")
9         match selection:
10             case 1:
11                 print("Enter the name of the class you wish to view.")
12                 class_name = input("=> ")
13                 view_class(class_name)
14             case 2:
15                 list_class()
16             case 3:
17                 add_class()
18             case 4:
19                 print("Which class would you like to expunge?")
20                 class_name = input("=> ").lower()
21                 remove_class(class_name)
22             case 5:
23                 print("Which class would you like to modify?")
24                 class_name = input("=> ").lower()
25                 modify_grade(class_name)
26             case 6:
27                 print("Enter the year of classes you wish to view")
28                 year = check_option(1, 10_000)
29                 print(
30                     """
31 Which GPA would you like to view?
32 1\tUnweighted
33 2\tWeighted
34
35 """
36                 )
37                 weighted = True if check_option(1, 3) == 2 else False
38                 print(f"{year_gpa(year, weighted):.2f}")
39             case 7:
40                 print(
41                     """
42 Which GPA would you like to view?
43 1\tUnweighted
44 2\tWeighted
45
46 """
47                 )
48                 weighted = True if check_option(1, 3) == 2 else False
49                 print(f"{cumulative_gpa(weighted):.2f}")
50             case 8:
51                 check_college(cumulative_gpa())
52             case 9:
53                 raise SystemExit("Thank you for using the program")
54             case 10:
55                 splash_screen()
56
57 def splash_screen():
58     print(
59         """
60 1\tView a class
61 2\tView class history
62 3\tAdd a class
63 4\tExpunge a class
64 5\tModify class grade
65 6\tView your GPA for a particular year
66 7\tView your cumulative GPA
67 8\tView the colleges you can attend with your GPA (Virginia)
68 9\tQuit the program
69 10\tRepeat this message
70
71 """
72     )
73
74 def check_option(lower_bounds: int, upper_bounds: int, prompt: str = "=> ") -> int:
75     while True:
76         user_input = input(prompt)
77         if user_input.isdigit():
78             user_input = int(user_input)
79             if user_input in range(lower_bounds, upper_bounds):
80                 return user_input
81     print("Please enter a valid option.")

```

```

82
83 def read_gpa() -> dict:
84     path = "../resources/gpa.json"
85     if not exists(path):
86         write_gpa({"class": {}, "total_classes": 0})
87     with open(path) as gpa_file:
88         return load(gpa_file)
89
90
91 def write_gpa(data: dict) -> None:
92     with open("../resources/gpa.json", "w") as gpa_file:
93         dump(data, gpa_file)
94
95
96 def letter_to_gpa(letter: str, class_type: str, weighted: bool = False) -> float:
97     match letter:
98         case "A":
99             gpa = 4.0
100         case "A-":
101             gpa = 3.7
102         case "B+":
103             gpa = 3.3
104         case "B":
105             gpa = 3.0
106         case "B-":
107             gpa = 2.7
108         case "C+":
109             gpa = 2.3
110         case "C":
111             gpa = 2.0
112         case "C-":
113             gpa = 1.7
114         case "D+":
115             gpa = 1.3
116         case "D":
117             gpa = 1.0
118         case "F":
119             gpa = 0.0
120     if gpa != 0.0 and weighted:
121         match class_type:
122             case "AP" | "DE":
123                 gpa += 1.0
124             case "HN":
125                 gpa += 0.5
126     return gpa
127
128
129 def add_class():
130     gpa_data = read_gpa()
131     print("What is the name of your class?")
132     class_name = input("=> ").lower()
133     print(
134         """
135 What is your letter grade?
136 1\tA
137 2\tA-
138 3\tB+
139 4\tB
140 5\tB-
141 6\tC+
142 7\tC
143 8\tC-
144 9\tD+
145 10\tD
146 11\tF
147         """
148     )
149     selection = check_option(1, 12)
150     match selection:
151         case 1:
152             letter_grade = "A"
153         case 2:
154             letter_grade = "A-"
155         case 3:
156             letter_grade = "B+"
157         case 4:
158             letter_grade = "B"
159         case 5:
160             letter_grade = "B-"
161         case 6:
162             letter_grade = "C+"

```

```

163         case 7:
164             letter_grade = "C"
165         case 8:
166             letter_grade = "C-"
167         case 9:
168             letter_grade = "D+"
169         case 10:
170             letter_grade = "D"
171         case 11:
172             letter_grade = "F"
173     print(
174         """
175 What type of class are you in?
176 1\tAdvanced Placement
177 2\tDual Enrollment
178 3\tHonors
179 4\tGeneral Education
180         """
181     )
182     selection = check_option(1, 5)
183     match selection:
184         case 1:
185             class_type = "AP"
186         case 2:
187             class_type = "DE"
188         case 3:
189             class_type = "HN"
190         case 4:
191             class_type = "GE"
192     print("What year did you take this class?")
193     year = check_option(0, 10_000)
194
195     gpa = letter_to_gpa(letter_grade, class_type)
196     weighted_gpa = letter_to_gpa(letter_grade, class_type, weighted=True)
197     gpa_data["class"][class_name] = {
198         "grade": letter_grade,
199         "type": class_type,
200         "year": year,
201         "gpa": gpa,
202         "weighted_gpa": weighted_gpa
203     }
204     gpa_data["total_classes"] += 1
205     write_gpa(gpa_data)
206
207
208 def remove_class(class_name: str) -> None:
209     gpa_data = read_gpa()
210     x = gpa_data["class"].pop(class_name)
211     if x:
212         gpa_data["total_classes"] -= 1
213     write_gpa(gpa_data)
214
215
216 def modify_grade(class_name: str):
217     gpa_data = read_gpa()
218     for key in gpa_data["class"]:
219         if key == class_name:
220             print(
221                 """
222 What is your letter grade?
223 1\tA
224 2\tA-
225 3\tB+
226 4\tB
227 5\tB-
228 6\tC+
229 7\tC
230 8\tC-
231 9\tD+
232 10\tD
233 11\tF
234                 """
235             )
236             selection = check_option(1, 12)
237             match selection:
238                 case 1:
239                     gpa_data["class"][key]["grade"] = "A"
240                 case 2:
241                     gpa_data["class"][key]["grade"] = "A-"
242                 case 3:
243                     gpa_data["class"][key]["grade"] = "B+"

```

```

244         case 4:
245             gpa_data["class"][key]["grade"] = "B"
246         case 5:
247             gpa_data["class"][key]["grade"] = "B-"
248         case 6:
249             gpa_data["class"][key]["grade"] = "C+"
250         case 7:
251             gpa_data["class"][key]["grade"] = "C"
252         case 8:
253             gpa_data["class"][key]["grade"] = "C-"
254         case 9:
255             gpa_data["class"][key]["grade"] = "D+"
256         case 10:
257             gpa_data["class"][key]["grade"] = "D"
258         case 11:
259             gpa_data["class"][key]["grade"] = "F"
260     gpa_data["class"][key]["gpa"] = letter_to_gpa(
261         gpa_data["class"][key]["grade"], gpa_data["class"][key]["grade"])
262     gpa_data["class"][key]["weighted gpa"] = letter_to_gpa(
263         gpa_data["class"][key]["grade"], gpa_data["class"][key]["grade"], True)
264     write_gpa(gpa_data)
265
266
267 def list_class() -> None:
268     gpa_data = read_gpa()["class"]
269     print()
270     for key in sorted(gpa_data.keys()):
271         print(key)
272     print()
273
274
275 def view_class(class_name):
276     gpa_data = read_gpa()["class"]
277     for key in gpa_data:
278         if key == class_name:
279             print(
280                 f"""
281 {key}
282 Grade: {gpa_data[key]['grade']}
283 Type: {gpa_data[key]['type']}
284 Year: {gpa_data[key]['year']}
285 GPA (Unweighted): {gpa_data[key]['gpa']}
286 GPA (Weighted): {gpa_data[key]['weighted gpa']}
287         """
288             )
289
290
291 def cumulative_gpa(weighted: bool = False) -> float:
292     gpa_data = read_gpa()
293     total_gpa = 0
294     total_weighted_gpa = 0
295     for key in gpa_data["class"]:
296         total_gpa += gpa_data["class"][key]["gpa"]
297         total_weighted_gpa += gpa_data["class"][key]["weighted gpa"]
298     if weighted:
299         return total_weighted_gpa / gpa_data["total classes"]
300     else:
301         return total_gpa / gpa_data["total classes"]
302
303
304 def year_gpa(year: int, weighted: bool = False) -> float:
305     gpa_data = read_gpa()["class"]
306     total_gpa = 0
307     total_weighted_gpa = 0
308     classes_in_year = 0
309     for key in gpa_data:
310         if gpa_data[key]["year"] == year:
311             total_gpa += gpa_data[key]["gpa"]
312             total_weighted_gpa += gpa_data[key]["weighted gpa"]
313             classes_in_year += 1
314     if classes_in_year == 0:
315         return 0
316     if weighted:
317         return total_weighted_gpa / classes_in_year
318     else:
319         return total_gpa / classes_in_year
320
321
322 def check_college(gpa):
323     # Data from https://www.ubridge.org/ubridgepvcc/guaranteed-admissions-universities/
324     colleges = {

```

```

325     "University of Virginia": 3.4,
326     "College of William & Mary": 3.6,
327     "Virginia Tech College of Arts and Sciences": 3.4,
328     "Virginia Tech College of Engineering": 3.2,
329     "Virginia Tech College of Agriculture & Life Sciences": 3.0,
330     "George Mason University": 2.85,
331     "Virginia Commonwealth University": 2.5,
332     "Regis University": 2.0,
333     "Shenandoah University": 2.5,
334     "Old Dominion University": 2.5,
335     "Mary Baldwin University": 2.5,
336     "Regent University": 2.5,
337     "Liberty University": 2.0,
338     "James Madison University": 3.0,
339     "Christopher Newport University": 3.5,
340     "Longwood University": 2.5,
341     "University of Mary Washington": 3.25,
342     "University of Lynchburg": 2.0,
343     "Radford University": 2.8,
344     "Ferrum University": 2.0,
345     "Bluefield University": 2.5,
346     "Virginia State University": 2.0,
347     "Troy University": 2.0,
348     "Norfolk State University": 2.0,
349     "ECPI University": 2.5,
350     "Strayer University": 2.0,
351     "Hollins University": 2.5,
352     "Randolph University": 2.7,
353     "Emory and Henry College": 2.5,
354     "University of Virginia's College at Wise": 2.5,
355     "Virginia Wesleyan College": 2.5,
356     "Virginia Union University": 2.0
357 }
358 colleges = dict(sorted(colleges.items()))
359 accepted_colleges = []
360 for key, value in colleges.items():
361     if gpa >= value:
362         accepted_colleges.append(key)
363 accepted_colleges = ", ".join(accepted_colleges)
364 if len(accepted_colleges) > 0:
365     print(f"You have the GPA required to attend:\n{accepted_colleges}")
366 else:
367     print("Your GPA is too low to attend any college in the state of Virginia.")
368
369
370 if __name__ == "__main__":
371     main()

```