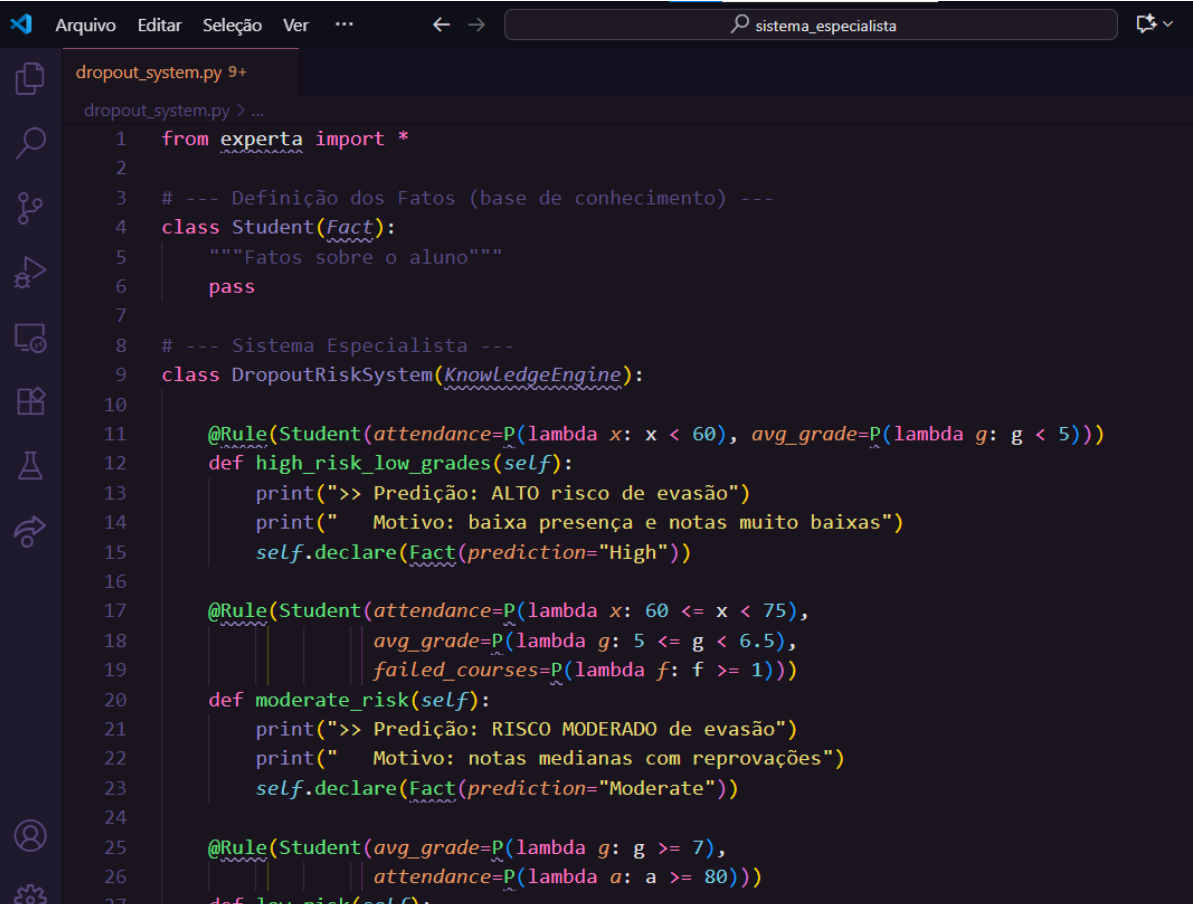


Questão 1

Documentação:

<https://colab.research.google.com/drive/1xkSNf-xosDydp6CmXFSSiNr9kHDzJwlh?usp=sharing>

Usamos o sistema PyKnow (expertsystem em Python) para montar o sistema baseado em regras:



```
Arquivo  Editar  Seleção  Ver  ...  sistema_especialista
dropout_system.py 9+
dropout_system.py > ...
1  from experta import *
2
3  # --- Definição dos Fatos (base de conhecimento) ---
4  class Student(Fact):
5      """Fatos sobre o aluno"""
6      pass
7
8  # --- Sistema Especialista ---
9  class DropoutRiskSystem(KnowledgeEngine):
10
11      @Rule(Student(attendance=P(lambda x: x < 60), avg_grade=P(lambda g: g < 5)))
12      def high_risk_low_grades(self):
13          print(">> Predição: ALTO risco de evasão")
14          print("  Motivo: baixa presença e notas muito baixas")
15          self.declare(Fact(prediction="High"))
16
17      @Rule(Student(attendance=P(lambda x: 60 <= x < 75),
18                  avg_grade=P(lambda g: 5 <= g < 6.5),
19                  failed_courses=P(lambda f: f >= 1)))
20      def moderate_risk(self):
21          print(">> Predição: RISCO MODERADO de evasão")
22          print("  Motivo: notas medianas com reprovações")
23          self.declare(Fact(prediction="Moderate"))
24
25      @Rule(Student(avg_grade=P(lambda g: g >= 7),
26                  attendance=P(lambda a: a >= 80)))
27      def low_risk(self):
```



```
Arquivo  Editar  Seleção  Ver  ...  ← →  sistema_especialista

dropout_system.py 9+
dropout_system.py > ...
9  class DropoutRiskSystem(KnowledgeEngine):
27      def low_risk(self):
28          print(">> Predição: BAIXO risco de evasão")
29          print("    Motivo: bom desempenho e frequência adequada")
30          self.declare(Fact(prediction="Low"))
31
32      @Rule(Student(last_term_change=P(lambda c: c <= -2)))
33      def sudden_drop(self):
34          print(">> Alerta adicional: queda brusca no desempenho")
35          self.declare(Fact(alert="Sudden drop"))
36
37      # --- Execução do sistema ---
38      if __name__ == "__main__":
39          engine = DropoutRiskSystem()
40
41          aluno1 = {
42              "attendance": 55,
43              "avg_grade": 4.5,
44              "failed_courses": 2,
45              "last_term_change": -1.0
46          }
47
48          print("=== Avaliando aluno 1 ===")
49          engine.reset()
50          engine.declare(Student(**aluno1))
51          engine.run()
52          print()
```

```
Arquivo  Editar  Seleção  Ver  ...  ← →  sistema_especialista

dropout_system.py 9+
dropout_system.py > ...
52      print()
53
54      aluno2 = {
55          "attendance": 70,
56          "avg_grade": 6.0,
57          "failed_courses": 1,
58          "last_term_change": -2.5
59      }
60
61      print("=== Avaliando aluno 2 ===")
62      engine.reset()
63      engine.declare(Student(**aluno2))
64      engine.run()
65      print()
66
67      aluno3 = {
68          "attendance": 90,
69          "avg_grade": 8.0,
70          "failed_courses": 0,
71          "last_term_change": 0.5
72      }
73
74      print("=== Avaliando aluno 3 ===")
75      engine.reset()
76      engine.declare(Student(**aluno3))
77      engine.run()
```

No terminal:

```
PROBLEMAS 19 SAÍDA CONSOLE DE DEPURAÇÃO TERMINAL PORTAS

>>
=== Avaliando aluno 1 ===
>> Predição: ALTO risco de evasão
    Motivo: baixa presença e notas muito baixas

=== Avaliando aluno 2 ===
>> Alerta adicional: queda brusca no desempenho
>> Predição: RISCO MODERADO de evasão
    Motivo: notas medianas com reprovações

=== Avaliando aluno 3 ===
>> Predição: BAIXO risco de evasão
    Motivo: bom desempenho e frequência adequada
```