

Insira seu RM: 123456
Insira sua idade: 19
Sua Participação foi autorizada, aluno de RM 123456!
Mais informações serão enviadas para seu e-mail cadastrado!
Informe a nota de satisfação (0-100): 93
Classificação: Qualidade

```
1
2  ~ import oracledb
3
4  # Codigo_Nivel_Atendimento_SAC_Melhores_Compras.py
5  from sys import excepthook, exception
6
7
8  ~ def classificar_satisfacao(nota): 1 usage
9  ~     """
10     Receber nota (int/float) entre 0 e 100 e retorna string classificando
11     - 'Qualidade' para nota > 90
12     - 'Neutro' para 70 <= nota <= 89
13     - 'Insatisfação' para nota < 70
14     Valida entrada e lança ValueError se inválida
15     """
16
17     try:
18         valor = float(nota)
19     except Exception:
20         raise ValueError("Nota inválida: informe um número entre 0 e 100.")
21
22     if valor >= 90:
23         return "Qualidade"
24     elif 70 <= valor <= 89:
25         return "Neutro"
26     else:
27         return "Insatisfatorio"
28
```

```

58
59     except Exception as e:
60         print(f" Erro ao salvar no banco: {e}")
61
62     finally:
63         if cursor:
64             cursor.close()
65         if conn:
66             conn.close()
67
68     def main(): 1 usage
69         rm = input(" Insira seu RM: ")
70         idade = input("Insira sua idade: ")
71
72         if int(idade) >= 18:
73             print("Sua Participação foi autorizada, aluno de RM {}".format(rm))
74             print("Mais informações serão enviadas para seu e-mail cadastrado!")
75
76             entrada = input("Informe a nota de satisfação (0-100): ").strip()
77             try:
78                 classificacao = classificar_satisfacao(entrada)
79                 print(f"Classificação: {classificacao}")
80             except ValueError as e:
81                 print(f"Erro: {e}")
82
83         else:
84             print("Infelizmente, a participação não foi autorizada, aluno de RM{}".format(rm))
85
86     if __name__ == "__main__":
87         main()

```

```

30 # Função para salvar no Oracle
31
32 def salvar_no_oracle(rm, idade, nota, classificacao):
33     """
34
35     Salva os dados de avaliação no banco de dados Oracle.
36     Ajuste as credenciais de conexão (user, password, dsn).
37     """
38
39     try:
40         conn = oracledb.connect(
41             user="SEU_USUARIO",          # TROQUE AQUI
42             password="SUA_SENHA",        # TROQUE AQUI
43             dsn="localhost:1521/XEPDB1"  # TROQUE AQUI (host:porta/ServiceName
44         )
45         cursor = conn.cursor()
46
47         # Exemplo: inserindo em SGV_SAC_CHAMADO
48         # Ajuste os campos conforme sua modelagem
49         cursor.execute( statement: """
50             INSERT INTO SGV_SAC_CHAMADO
51                 (cd_chamado, cd_cliente, vl_indice_satisfacao, st_chamado)
52             VALUES
53                 (SGV_SAC_CHAMADO_SEQ.NEXTVAL, :1, :2,:3)
54             """, parameters: (rm, nota, classificacao))
55
56         conn.commit()
57         print("✓ Avaliação salva no bano Oracle com sucesso!")
58

```