

```

1 from model.conexaoDB import ConexaoDB
2 from model.turmaModel import Turma
3
4 class TurmaDAO:
5     __slots__ = (
6         '_con'
7     )
8
9     def __init__(self):
10         self._con = ConexaoDB.conectar()
11
12     def inserirTurma(self, turma):
13         """
14         Adiciona uma turma ao banco de dados
15         :param turma: Espera um objeto do tipo turma
16         :return: True caso a turma seja adicionada e False caso contrario
17         """
18         sql = "INSERT INTO Turma(nome,turno) VALUES (?,?);"
19         valores = (turma.nome, turma.turno)
20         res = ConexaoDB.executarSql(sql,valores)
21         return res == 1
22
23     def atualizarTurma(self, turma):
24         """
25         Atualiza uma turma no banco de dados
26         :param turma: Espera um objeto do tipo turma
27         :return: True caso a turma seja atualizada e False caso contrario
28         """
29         sql = "UPDATE Turma SET nome=?, turno=? WHERE id=?;"
30         valores = (turma.nome, turma.turno, turma.id)
31         res = ConexaoDB.executarSql(sql,valores)
32         return res == 1
33
34     def excluirTurma(self, id):
35         """
36         Exclui uma turma do banco de dados
37         :param id: Espera o id(string) da turma a ser excluída
38         :return: True caso a turma seja excluída e False caso contrario
39         """
40         sql = "DELETE FROM turma WHERE id = " + str(id)
41         cursor = self._con.cursor()
42         cursor.execute(sql)
43         self._con.commit()
44         res = cursor.rowcount
45         return res == 1
46
47     def buscarTurma(self,id):
48         """
49         Busca uma turma no banco de dados
50         :param id: Espera o id da turma a ser buscada
51         :return: A turma de acordo com o id informado
52         """
53         try:
54             sql = "SELECT id,nome,turno FROM turma WHERE id =" + str(id) + ";"
55             cursor = self._con.cursor()
56             cursor.execute(sql)
57             res = cursor.fetchone()
58             turma = Turma(res[0], res[1], res[2])
59             return turma
60         except Exception as e:

```

```

61         print(str(e))
62         return None
63
64     def buscarTurmaPorNome(self, nome):
65         """
66         Busca uma turma no banco de dados pelo seu nome
67         :param nome: Espera o nome da turma a ser buscada
68         :return: A turma de acordo com o nome informado
69         """
70         try:
71             sql = "SELECT id,nome,turno FROM turma WHERE nome = '" + nome +
72             ";"
73             cursor = self._con.cursor()
74             cursor.execute(sql)
75             res = cursor.fetchone()
76             turma = Turma(res[0], res[1], res[2])
77             return turma
78         except Exception as e:
79             print(str(e))
80             return None
81
82     def buscarTurmas(self, inicio=0, quant=100):
83         """
84         Busca as turmas do banco de dados
85         :param quant: Espera a quantidade de turmas a serem buscadas
86         :return: diversas Turmas de acordo com a quantidade informada
87         """
88         turmas = []
89         try:
90             sql = "SELECT id,nome,turno FROM turma"
91             #sql += " LIMIT " + str(inicio)
92             #sql += "," + str(quant)+";"
93             cursor = self._con.cursor()
94             cursor.execute(sql)
95             res = cursor.fetchmany(quant)
96             turmas = self._montarResultado(res)
97             return turmas
98         except Exception as e:
99             print(e)
100         return turmas
101
102     def _montarResultado(self, res):
103         turmas = []
104         for linha in res:
105             turma = Turma(linha[0], linha[1], linha[2])
106             turmas.append(turma)
107         return turmas
108

```