

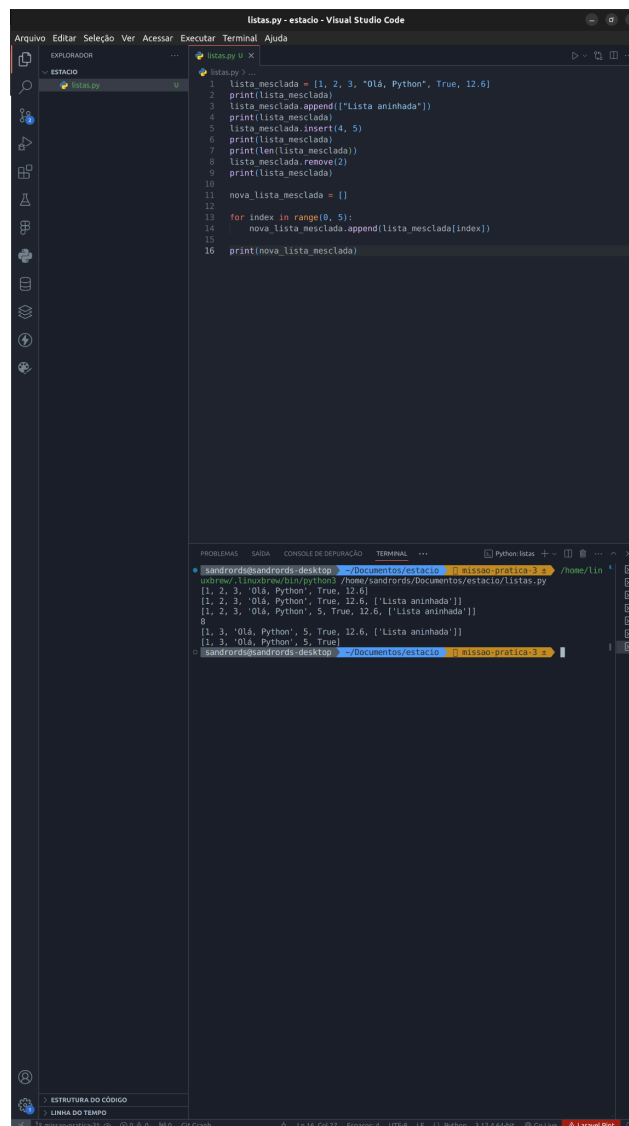
Missão Prática 2

Execução do Projeto

Configurações da Máquina

A atividade foi executada através de um desktop com o sistema operacional Ubuntu 22.04 LTS.

1. Microatividade 1



```
listas.py - estacio - Visual Studio Code

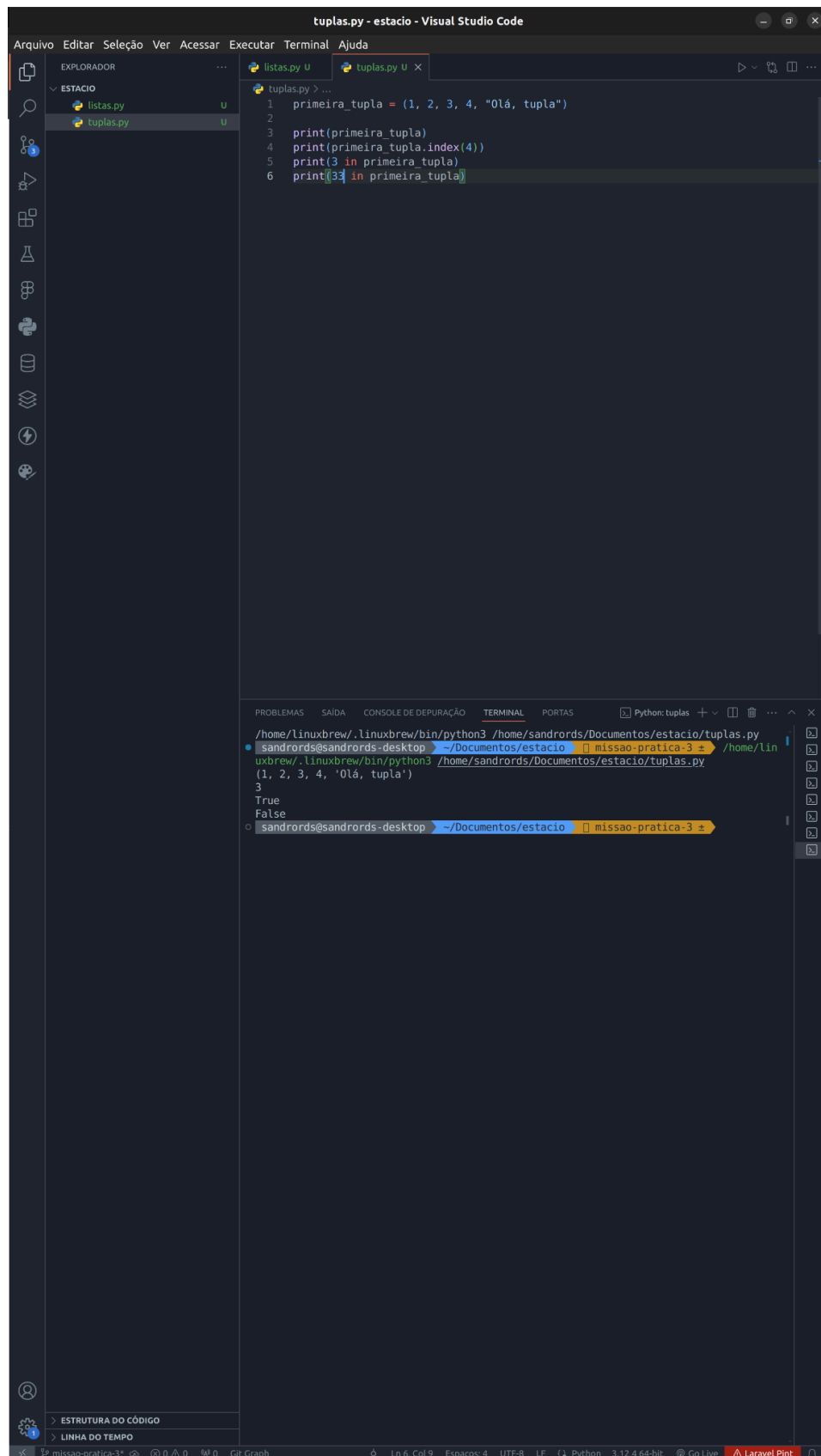
1 lista_mesclada = [1, 2, 3, 'Olá, Python', True, 12.6]
2 print(lista_mesclada)
3 lista_mesclada.append(['Lista aninhada'])
4 print(lista_mesclada)
5 lista_mesclada.insert(4, 5)
6 print(lista_mesclada)
7 print(len(lista_mesclada))
8 lista_mesclada.remove(2)
9 print(lista_mesclada)
10
11 nova_lista_mesclada = []
12
13 for index in range(0, 5):
14     nova_lista_mesclada.append(lista_mesclada[index])
15
16 print(nova_lista_mesclada)
```

```

sandrords@sandrords-desktop: ~/Documentos/estacio - missao-pratica-3 $ python3 /home/sandrords/.linuxbrew/bin/python3 /home/sandrords/Documentos/estacio/listas.py
[1, 2, 3, 'Olá, Python', True, 12.6]
[1, 2, 3, 'Olá, Python', True, 12.6, ['Lista aninhada']]
[1, 2, 3, 'Olá, Python', 5, True, 12.6, ['Lista aninhada']]
8
[1, 3, 'Olá, Python', 5, True, 12.6, ['Lista aninhada']]
[1, 3, 'Olá, Python', 5, True]
sandrords@sandrords-desktop: ~/Documentos/estacio - missao-pratica-3 $
```

Criando script testando a manipulação de Listas.

2. Microatividade 2



The image shows a Visual Studio Code editor window titled "tuplas.py - estacio - Visual Studio Code". The Explorer sidebar on the left shows a folder named "ESTACIO" containing two files: "listas.py" and "tuplas.py". The "tuplas.py" file is open in the editor, showing the following Python code:

```
1 primeira_tupla = (1, 2, 3, 4, "Olá, tupla")
2
3 print(primeira_tupla)
4 print(primeira_tupla.index(4))
5 print(3 in primeira_tupla)
6 print(33 in primeira_tupla)
```

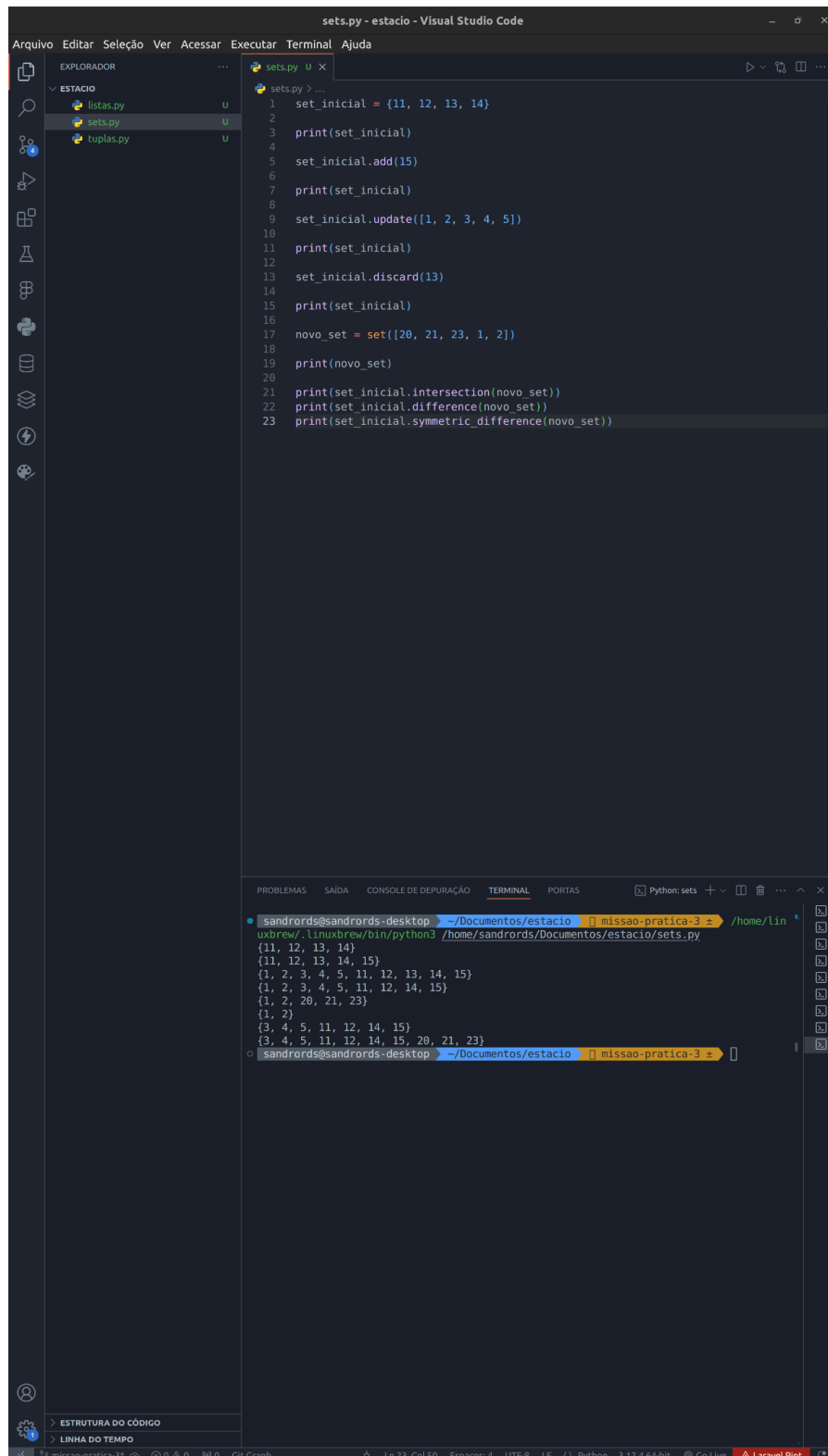
The bottom panel of the editor shows the "TERMINAL" output, which displays the execution of the script:

```
/home/linuxbrew/.linuxbrew/bin/python3 /home/sandrords/Documents/estacio/tuplas.py
sandrords@sandrords-desktop ~/Documentos/estacio missao-pratica-3
/home/linuxbrew/.linuxbrew/bin/python3 /home/sandrords/Documents/estacio/tuplas.py
(1, 2, 3, 4, 'Olá, tupla')
3
True
False
sandrords@sandrords-desktop ~/Documentos/estacio missao-pratica-3
```

The status bar at the bottom indicates the file is "missao-pratica-3*", the encoding is "UTF-8", the line and column are "Ln 6, Col 9", and the Python interpreter is "Python 3.12.4 64-bit".

Criando script testando a manipulação de Tuplas.

3. Microatividade 3



The image shows a Visual Studio Code editor window titled "sets.py - estacio - Visual Studio Code". The Explorer sidebar on the left shows a folder named "ESTACIO" containing three files: "listas.py", "sets.py", and "tuplas.py". The "sets.py" file is open in the editor, showing the following Python code:

```
1 set_inicial = {11, 12, 13, 14}
2
3 print(set_inicial)
4
5 set_inicial.add(15)
6
7 print(set_inicial)
8
9 set_inicial.update([1, 2, 3, 4, 5])
10
11 print(set_inicial)
12
13 set_inicial.discard(13)
14
15 print(set_inicial)
16
17 novo_set = set([20, 21, 23, 1, 2])
18
19 print(novo_set)
20
21 print(set_inicial.intersection(novo_set))
22 print(set_inicial.difference(novo_set))
23 print(set_inicial.symmetric_difference(novo_set))
```

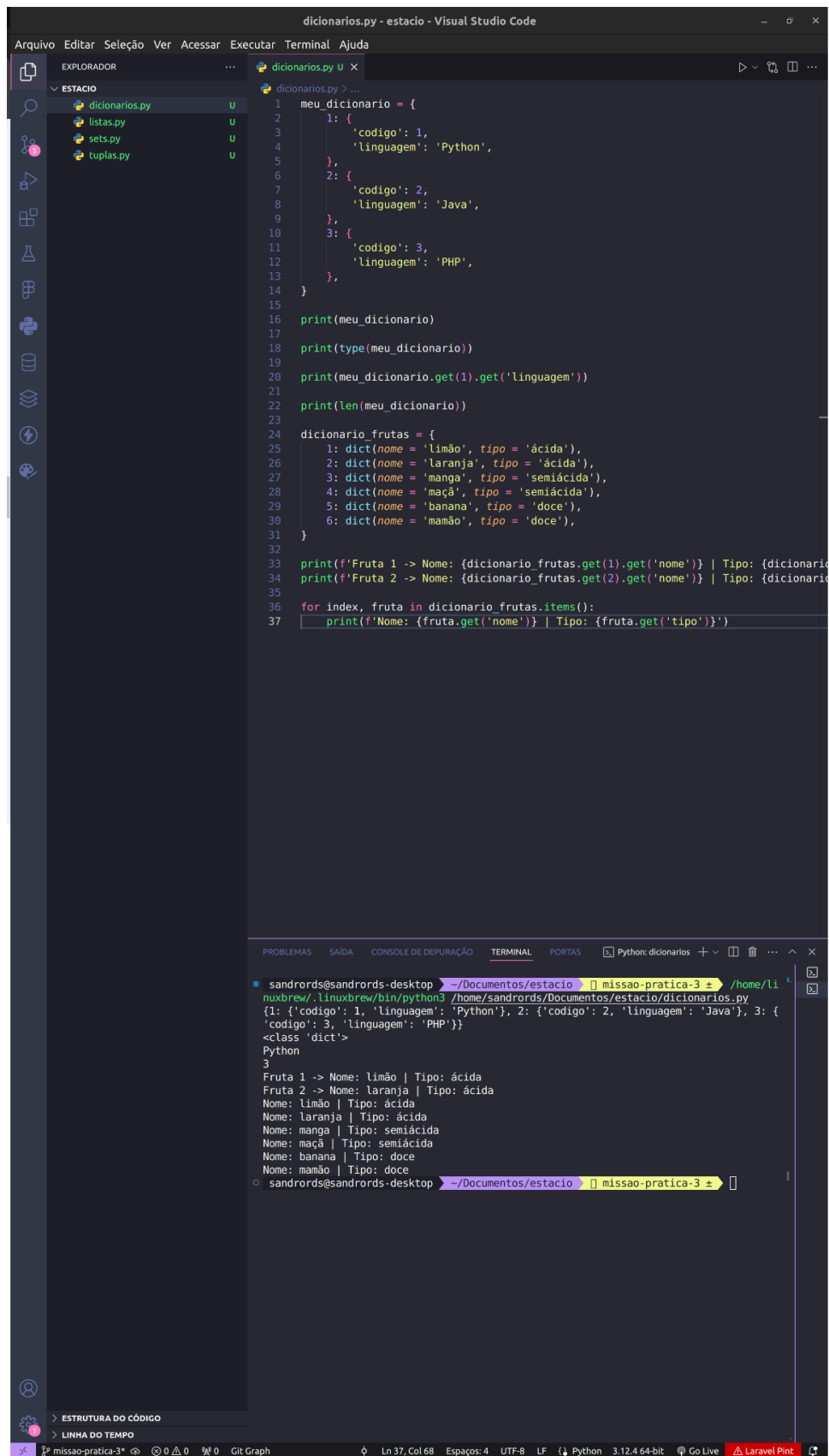
The TERMINAL panel at the bottom shows the output of running the script. The command executed is `python3 /home/sandrords/Documents/estacio/sets.py`. The output is as follows:

```
python3 /home/sandrords/Documents/estacio/sets.py
{11, 12, 13, 14}
{11, 12, 13, 14, 15}
{1, 2, 3, 4, 5, 11, 12, 13, 14, 15}
{1, 2, 3, 4, 5, 11, 12, 14, 15}
{1, 2, 20, 21, 23}
{1, 2}
{3, 4, 5, 11, 12, 14, 15}
{3, 4, 5, 11, 12, 14, 15, 20, 21, 23}
{1, 2, 20, 21, 23}
```

The status bar at the bottom indicates the file is "sets.py" (Ln 23, Col 50) with a file size of 4 bytes, using UTF-8 encoding and LF line endings. The Python interpreter is set to "Python 3.12.4 64-bit".

Criando script testando a manipulação de Sets.

4. Microatividade 4



The image shows a Visual Studio Code editor window with a Python file named `dicionarios.py` open. The file contains two dictionaries and several print statements to test their manipulation.

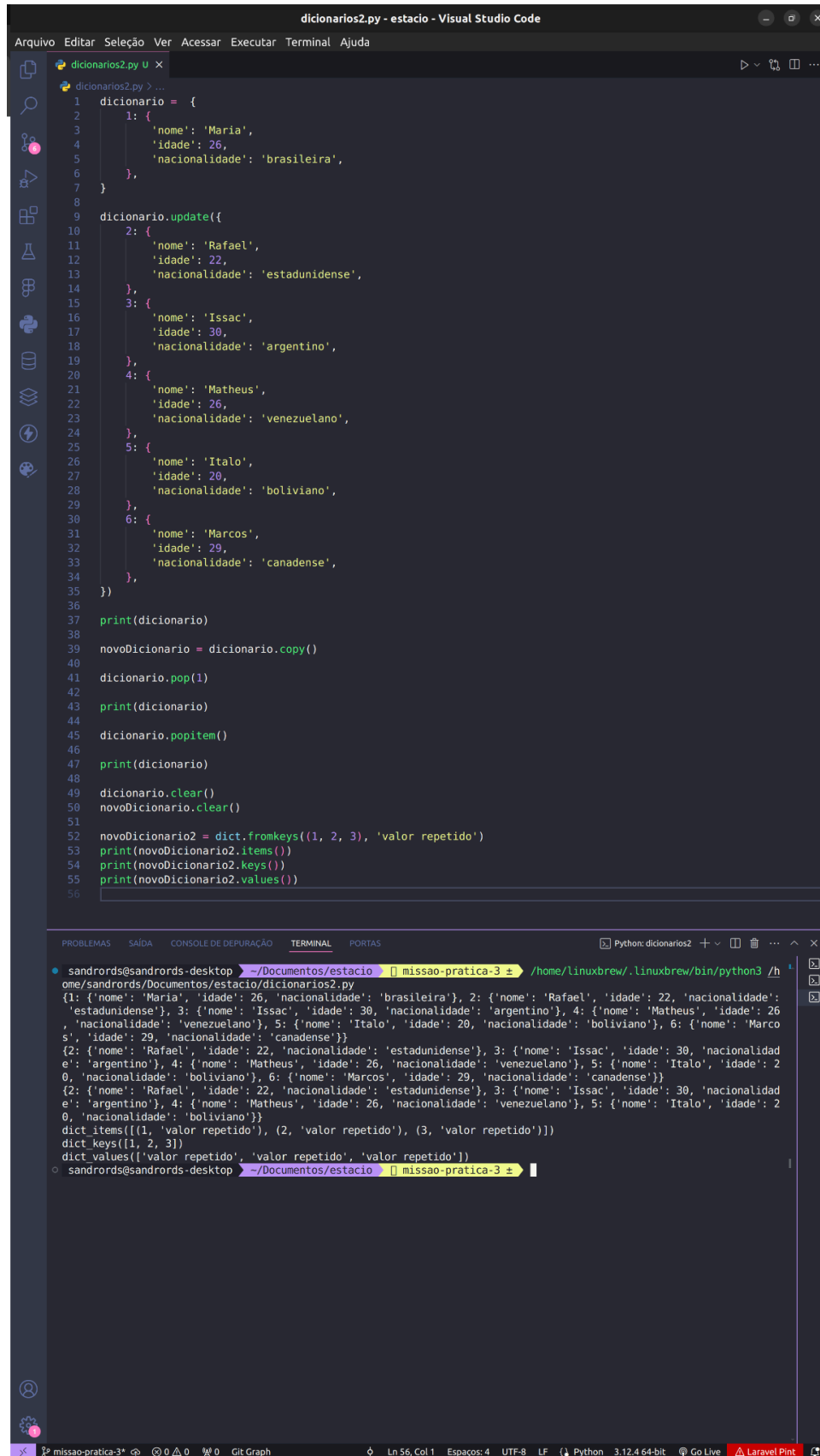
```
1 meu_dicionario = {
2     1: {
3         'codigo': 1,
4         'linguagem': 'Python',
5     },
6     2: {
7         'codigo': 2,
8         'linguagem': 'Java',
9     },
10    3: {
11        'codigo': 3,
12        'linguagem': 'PHP',
13    },
14 }
15
16 print(meu_dicionario)
17
18 print(type(meu_dicionario))
19
20 print(meu_dicionario.get(1).get('linguagem'))
21
22 print(len(meu_dicionario))
23
24 dicionario_frutas = {
25     1: dict(nome = 'limão', tipo = 'ácida'),
26     2: dict(nome = 'laranja', tipo = 'ácida'),
27     3: dict(nome = 'manga', tipo = 'semiácida'),
28     4: dict(nome = 'maçã', tipo = 'semiácida'),
29     5: dict(nome = 'banana', tipo = 'doce'),
30     6: dict(nome = 'mamão', tipo = 'doce'),
31 }
32
33 print(f'Fruta 1 -> Nome: {dicionario_frutas.get(1).get('nome')} | Tipo: {dicionario_frutas.get(1).get('tipo')}')
34 print(f'Fruta 2 -> Nome: {dicionario_frutas.get(2).get('nome')} | Tipo: {dicionario_frutas.get(2).get('tipo')}')
35
36 for index, fruta in dicionario_frutas.items():
37     print(f'Nome: {fruta.get('nome')} | Tipo: {fruta.get('tipo')}')
```

The terminal output shows the execution of the script:

```
sandrords@sandrords-desktop ~/Documentos/estacio [missao-pratica-3] /home/li
nuxbrew/.linuxbrew/bin/python3 /home/sandrords/Documentos/estacio/dicionarios.py
{1: {'codigo': 1, 'linguagem': 'Python'}, 2: {'codigo': 2, 'linguagem': 'Java'}, 3: {'codigo': 3, 'linguagem': 'PHP'}}
<class 'dict'>
Python
3
Fruta 1 -> Nome: limão | Tipo: ácido
Fruta 2 -> Nome: laranja | Tipo: ácido
Nome: limão | Tipo: ácido
Nome: laranja | Tipo: ácido
Nome: manga | Tipo: semiácida
Nome: maçã | Tipo: semiácida
Nome: banana | Tipo: doce
Nome: mamão | Tipo: doce
sandrords@sandrords-desktop ~/Documentos/estacio [missao-pratica-3]
```

Criando script testando a manipulação de Dicionários.

5. Microatividade 5



The image shows a Visual Studio Code editor window titled 'dicionarios2.py - estacio - Visual Studio Code'. The editor contains a Python script that creates a dictionary, updates it, and performs various operations like copying, popping, and clearing. The terminal at the bottom shows the execution of the script, displaying the dictionary's state at different points and the results of the operations.

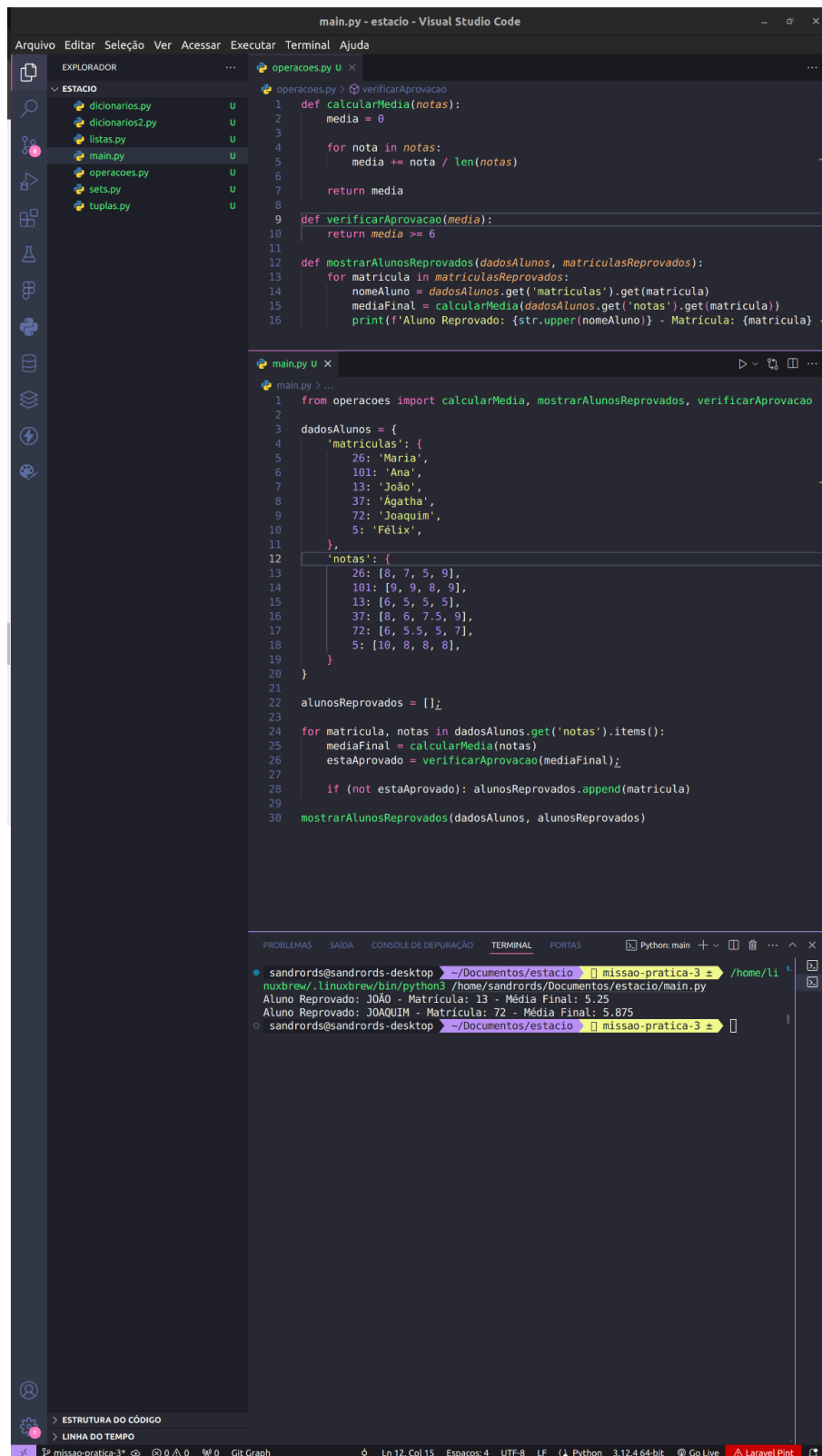
```
1 dicionario = {
2     1: {
3         'nome': 'Maria',
4         'idade': 26,
5         'nacionalidade': 'brasileira',
6     },
7 }
8
9 dicionario.update({
10     2: {
11         'nome': 'Rafael',
12         'idade': 22,
13         'nacionalidade': 'estadunidense',
14     },
15     3: {
16         'nome': 'Issac',
17         'idade': 30,
18         'nacionalidade': 'argentino',
19     },
20     4: {
21         'nome': 'Matheus',
22         'idade': 26,
23         'nacionalidade': 'venezuelano',
24     },
25     5: {
26         'nome': 'Italo',
27         'idade': 20,
28         'nacionalidade': 'boliviano',
29     },
30     6: {
31         'nome': 'Marcos',
32         'idade': 29,
33         'nacionalidade': 'canadense',
34     },
35 })
36
37 print(dicionario)
38
39 novoDicionario = dicionario.copy()
40
41 dicionario.pop(1)
42
43 print(dicionario)
44
45 dicionario.popitem()
46
47 print(dicionario)
48
49 dicionario.clear()
50 novoDicionario.clear()
51
52 novoDicionario2 = dict.fromkeys((1, 2, 3), 'valor repetido')
53 print(novoDicionario2.items())
54 print(novoDicionario2.keys())
55 print(novoDicionario2.values())
56
```

The terminal output shows the following:

```
Python: dicionarios2
sandrrds@sandrrds-desktop ~/Documentos/estacio missao-pratica-3 $ /home/linuxbrew/.linuxbrew/bin/python3 /h
.../Documentos/estacio/dicionarios2.py
{1: {'nome': 'Maria', 'idade': 26, 'nacionalidade': 'brasileira'}, 2: {'nome': 'Rafael', 'idade': 22, 'nacionalidad
e': 'estadunidense'}, 3: {'nome': 'Issac', 'idade': 30, 'nacionalidade': 'argentino'}, 4: {'nome': 'Matheus', 'idade': 26
, 'nacionalidade': 'venezuelano'}, 5: {'nome': 'Italo', 'idade': 20, 'nacionalidade': 'boliviano'}, 6: {'nome': 'Marco
s', 'idade': 29, 'nacionalidade': 'canadense'}}
{2: {'nome': 'Rafael', 'idade': 22, 'nacionalidade': 'estadunidense'}, 3: {'nome': 'Issac', 'idade': 30, 'nacionalidad
e': 'argentino'}, 4: {'nome': 'Matheus', 'idade': 26, 'nacionalidade': 'venezuelano'}, 5: {'nome': 'Italo', 'idade': 2
0, 'nacionalidade': 'boliviano'}, 6: {'nome': 'Marcos', 'idade': 29, 'nacionalidade': 'canadense'}}
{2: {'nome': 'Rafael', 'idade': 22, 'nacionalidade': 'estadunidense'}, 3: {'nome': 'Issac', 'idade': 30, 'nacionalidad
e': 'argentino'}, 4: {'nome': 'Matheus', 'idade': 26, 'nacionalidade': 'venezuelano'}, 5: {'nome': 'Italo', 'idade': 2
0, 'nacionalidade': 'boliviano'}}
dict_items([(1, 'valor repetido'), (2, 'valor repetido'), (3, 'valor repetido')])
dict_keys([1, 2, 3])
dict_values(['valor repetido', 'valor repetido', 'valor repetido'])
sandrrds@sandrrds-desktop ~/Documentos/estacio missao-pratica-3 $
```

Criando script explorando ainda mais os recursos dos Dicionários.

Missão Prática



The screenshot shows the Visual Studio Code interface with two Python files open. The file explorer on the left shows a project named 'ESTACIO' with files: dicionarios.py, dicionarios2.py, listas.py, main.py, operacoes.py, sets.py, and tuplas.py. The 'operacoes.py' file contains three functions: `calcularMedia(notas)` which calculates the average of a list of grades, `verificarAprovacao(media)` which checks if the average is greater than or equal to 6, and `mostrarAlunosReprovados(dadosAlunos, matriculasReprovados)` which iterates through student data and prints the names and IDs of students who failed. The 'main.py' file imports these functions and defines a dictionary `dadosAlunos` with student IDs, names, and lists of grades. It then iterates through this data, calculates the average for each student, checks if they passed, and appends failing students to a list. Finally, it calls `mostrarAlunosReprovados` to display the results. The terminal at the bottom shows the execution output, identifying João and Joaquim as failing students.

```
main.py - estacio - Visual Studio Code

Arquivo  Editar  Seleção  Ver  Acessar  Executar  Terminal  Ajuda

EXPLORADOR
ESTACIO
  dicionarios.py
  dicionarios2.py
  listas.py
  main.py
  operacoes.py
  sets.py
  tuplas.py

operacoes.py
1 def calcularMedia(notas):
2     media = 0
3
4     for nota in notas:
5         media += nota / len(notas)
6
7     return media
8
9 def verificarAprovacao(media):
10     return media >= 6
11
12 def mostrarAlunosReprovados(dadosAlunos, matriculasReprovados):
13     for matricula in matriculasReprovados:
14         nomeAluno = dadosAlunos.get('matriculas').get(matricula)
15         mediaFinal = calcularMedia(dadosAlunos.get('notas').get(matricula))
16         print(f'Aluno Reprovado: {str.upper(nomeAluno)} - Matrícula: {matricula}')
```

```
main.py
1 from operacoes import calcularMedia, mostrarAlunosReprovados, verificarAprovacao
2
3 dadosAlunos = {
4     'matriculas': {
5         26: 'Maria',
6         101: 'Ana',
7         13: 'João',
8         37: 'Agatha',
9         72: 'Joaquim',
10        5: 'Félix',
11    },
12    'notas': {
13        26: [8, 7, 5, 9],
14        101: [9, 9, 8, 9],
15        13: [6, 5, 5, 5],
16        37: [8, 6, 7.5, 9],
17        72: [6, 5.5, 5, 7],
18        5: [10, 8, 8, 8],
19    }
20 }
21
22 alunosReprovados = []
23
24 for matricula, notas in dadosAlunos.get('notas').items():
25     mediaFinal = calcularMedia(notas)
26     estaAprovado = verificarAprovacao(mediaFinal)
27
28     if (not estaAprovado): alunosReprovados.append(matricula)
29
30 mostrarAlunosReprovados(dadosAlunos, alunosReprovados)
```

```
TERMINAL
Python: main
sandrords@sandrords-desktop ~/Documentos/estacio missao-pratica-3
nuxbrew/.linuxbrew/bin/python3 /home/sandrords/Documentos/estacio/main.py
Aluno Reprovado: JOÃO - Matrícula: 13 - Média Final: 5.25
Aluno Reprovado: JOAQUIM - Matrícula: 72 - Média Final: 5.875
sandrords@sandrords-desktop ~/Documentos/estacio missao-pratica-3
```

Criando o script para mostrar os alunos reprovados