

1.1 Question 1

En générant un rapport de couverture de code à l'aide de l'IDE PyCharm, quelle est la couverture totale du code (statement coverage, missing, excluded, coverage %) pour Requests v2.28.1 ?

La couverture totale du code est de 88%.

Coverage report: 88%					Filter...	
coverage.py v7.1.0, created at 2023-02-02 14:27 -0500						
Module	statements	missing	excluded	coverage		
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\jb_pytest_runner.py	33	3	0	91%		
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\jb_runner_tools.py	173	48	0	72%		
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\jb_serial_tree_manager.py	56	7	0	88%		
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\jb_utils.py	71	34	0	52%		
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\teamcity__init__.py	6	0	0	100%		
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\teamcity\common.py	98	60	0	39%		
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\teamcity\diff_tools.py	62	36	0	42%		
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\teamcity\jb_local_exc_store.py	11	3	0	73%		
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\teamcity\messages.py	153	58	0	62%		
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\teamcity\pytest_plugin.py	294	123	0	58%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests__init__.py	68	24	0	65%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests__version__.py	10	0	0	100%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests_internal_utils.py	19	0	0	100%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\adapters.py	216	14	0	94%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\api.py	19	3	0	84%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\auth.py	173	21	0	88%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\certs.py	4	1	0	75%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\compat.py	30	2	0	93%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\cookies.py	239	53	0	78%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\exceptions.py	37	0	0	100%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\help.py	62	19	0	69%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\hooks.py	14	0	0	100%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\models.py	455	33	0	93%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\packages.py	17	0	0	100%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\sessions.py	268	12	0	96%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\status_codes.py	14	0	0	100%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\structures.py	39	0	0	100%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\utils.py	462	68	0	86%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests__init__.py	6	2	0	67%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\compat.py	12	2	0	83%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\conftest.py	37	3	0	92%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_help.py	12	0	0	100%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_hooks.py	9	0	0	100%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_lowlevel.py	227	0	0	100%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_packages.py	7	0	0	100%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_requests.py	1651	41	0	98%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_structures.py	42	0	0	100%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_testserver.py	107	14	0	87%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_utils.py	308	5	0	98%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\testserver__init__.py	0	0	0	100%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\testserver\server.py	86	4	0	95%		
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\utils.py	13	1	0	92%		
Total	5640	694	0	88%		
coverage.py v7.1.0, created at 2023-02-02 14:27 -0500						

1.2 Question 2

En générant un nouveau rapport de couverture de code à l'aide de l'IDE PyCharm, quelle est la couverture totale du code (files, lines, statements, missing, excluded, branches, partial coverage %) pour Requests v2.28.1 ? Y a-t-il une différence avec le rapport précédent ? Expliquez pourquoi il y a/n'y a pas de différence.

La couverture totale du code est de 85%. Cette couverture est inférieure à la précédente car les tests n'exécutent pas toutes les possibilités (true/false) des branches.

[illegible]

1.3 Question 3

Vous devez exclure un fichier de code source du processus de génération de rapport de test. Quel fichier avez-vous exclu (nom et répertoire du fichier) ? Comment l'avez-vous exclu du processus ? Comment les résultats changent-ils par rapport au dernier rapport (conserver l'option pour avoir la couverture des branches) ?

Nous avons exclu le fichier tests/test_help.py par la commande :

« python -m coverage run -m --branch --omit tests/test_help.py -m pytest »

La couverture de code est supérieure à celle du dernier rapport, elle a augmenté de 6%.

Coverage report: 91%

coverage.py v7.1.0, created at 2023-02-02 16:06 -0500

Module	statements	missing	excluded	branches	partial	coverage
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests_init_.py	68	24	0	10	4	62%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests_version_.py	10	0	0	0	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests_internal_utils.py	19	0	0	2	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\adapters.py	216	14	0	78	7	93%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\api.py	19	3	0	0	0	84%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\auth.py	173	21	0	60	15	84%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\certs.py	4	1	0	2	1	67%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\compat.py	30	2	0	2	1	91%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\cookies.py	239	53	0	98	9	72%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\exceptions.py	37	0	0	2	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\help.py	62	19	0	18	5	60%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\hooks.py	14	0	0	10	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\models.py	455	33	0	188	19	92%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\packages.py	17	0	0	10	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\sessions.py	268	12	0	100	5	95%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\status_codes.py	14	0	0	10	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\structures.py	39	0	0	8	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\utils.py	482	68	0	214	12	86%
__init__.py	6	2	0	0	0	67%
compat.py	12	2	0	0	0	83%
conftest.py	37	3	0	0	0	92%
test_help.py	12	0	0	0	0	100%
test_hooks.py	9	0	0	2	0	100%
test_lowlevel.py	227	0	0	8	0	100%
test_packages.py	7	0	0	0	0	100%
test_requests.py	1651	38	0	78	6	97%
test_structures.py	42	0	0	0	0	100%
test_testserver.py	107	14	0	2	0	87%
test_utils.py	308	5	0	18	4	97%
testserver_init_.py	0	0	0	0	0	100%
testserver\server.py	86	4	0	18	2	94%
utils.py	13	1	0	4	1	88%
Total	4683	319	0	942	91	91%

coverage.py v7.1.0, created at 2023-02-02 16:06 -0500

1.4 Question 4

Veillez expliquer, étape par étape, comment utiliser coverage.py pour calculer la couverture de test de Requests sans IDE (utiliser l'option pour branch coverage).

Nous avons utilisé coverage.py pour rouler la couverture de tests par la commande :

« python -m coverage run -m --branch -m pytest »

Nous avons généré le rapport de couverture de tests par la commande :

« python -m coverage html »

Coverage report: 91%

coverage.py v7.10, created at 2023-02-02 16:12 -0500

Module	statements	missing	excluded	branches	partial	coverage
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests_init_.py	68	24	0	10	4	62%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests_version_.py	10	0	0	0	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests_internal_utils.py	19	0	0	2	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\adapters.py	216	14	0	78	7	93%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\api.py	19	3	0	0	0	84%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\auth.py	173	21	0	60	15	84%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\certs.py	4	1	0	2	1	67%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\compat.py	30	2	0	2	1	91%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\cookies.py	239	53	0	98	9	72%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\exceptions.py	37	0	0	2	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\help.py	62	19	0	18	5	60%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\hooks.py	14	0	0	10	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\models.py	455	33	0	188	19	92%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\packages.py	17	0	0	10	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\sessions.py	268	12	0	100	5	95%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\status_codes.py	14	0	0	10	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\structures.py	39	0	0	8	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\utils.py	482	68	0	214	12	86%
__init__.py	6	2	0	0	0	67%
compat.py	12	2	0	0	0	83%
conftest.py	37	3	0	0	0	92%
test_help.py	12	0	0	0	0	100%
test_hooks.py	9	0	0	2	0	100%
test_lowlevel.py	227	0	0	8	0	100%
test_packages.py	7	0	0	0	0	100%
test_requests.py	1651	38	0	78	6	97%
test_structures.py	42	0	0	0	0	100%
test_testserver.py	107	14	0	2	0	87%
test_utils.py	308	5	0	18	4	97%
testserver_init_.py	0	0	0	0	0	100%
testserver\server.py	86	4	0	18	2	94%
utils.py	13	1	0	4	1	88%
Total	4683	319	0	942	91	91%

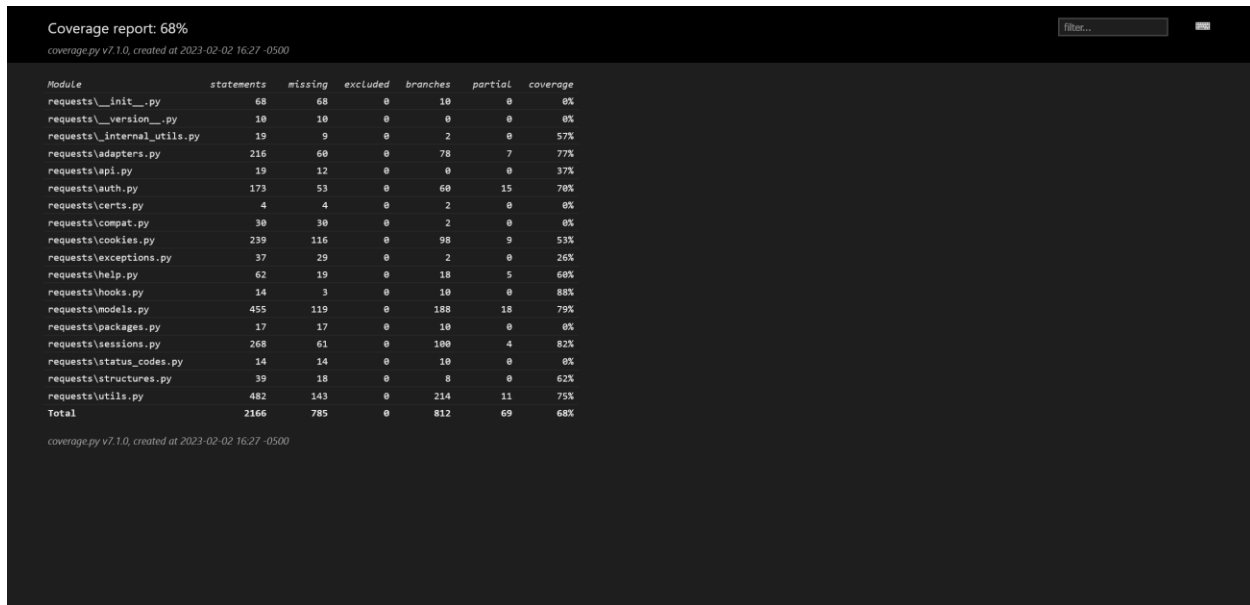
coverage.py v7.10, created at 2023-02-02 16:12 -0500

1.5 Question 5

Veillez expliquer, étape par étape, comment utiliser pytest pour calculer la couverture de test de Requests sans IDE (utiliser l'option pour branch coverage).

Nous avons utilisé pytest pour rouler la couverture de tests par la commande :
« `python -m pytest --cov=requests --cov-branch --cov-report=html tests/` »

Nous avons généré le rapport de couverture de tests par la commande :
« `python -m coverage html` »



Coverage report: 68%

coverage.py v7.1.0, created at 2023-02-02 16:27 -0500

Module	statements	missing	excluded	branches	partial	coverage
requests__init__.py	68	68	0	10	0	0%
requests__version__.py	10	10	0	0	0	0%
requests_internal_utils.py	19	9	0	2	0	57%
requests\adapters.py	216	60	0	78	7	77%
requests\api.py	19	12	0	0	0	37%
requests\auth.py	173	53	0	60	15	70%
requests\certs.py	4	4	0	2	0	0%
requests\compat.py	30	30	0	2	0	0%
requests\cookies.py	239	116	0	98	9	53%
requests\exceptions.py	37	29	0	2	0	26%
requests\help.py	62	19	0	18	5	60%
requests\hooks.py	14	3	0	10	0	88%
requests\models.py	455	119	0	188	18	79%
requests\packages.py	17	17	0	10	0	0%
requests\sessions.py	268	61	0	100	4	82%
requests\status_codes.py	14	14	0	10	0	0%
requests\structures.py	39	18	0	8	0	62%
requests\utils.py	482	143	0	214	11	75%
Total	2166	785	0	812	69	68%

coverage.py v7.1.0, created at 2023-02-02 16:27 -0500

1.6 Question 6

Vous devez créer deux nouveaux tests pour augmenter la couverture de test de Requests. Quels tests avez-vous écrit ? En utilisant l'une des deux méthodes de couverture de test des questions précédentes (PyCharm ou Coverage.py ou pytest), montrez quels critère(s) de couverture de test vos tests ont augmenté.

Nous avons effectué deux tests pour le fichier `utils.py`. Le premier test vérifie si la conversion entre un dictionnaire et une liste fonctionne lorsque l'on passe un dictionnaire. Le deuxième test vérifie si le dictionnaire est en ordre alphabétique en fonction de la clé (l'équipe).

Cela a fait augmenter la couverture de test du fichier `utils.py` de 86% à 87% si l'on compare avec la question 2.

```
new *
@pytest.mark.parametrize(
    "value, expected",
    (
        ({ "name": "Canadiens", "city": "Montreal"},
         {"name": "Canadiens", "city": "Montreal"}.items(),
         ("name", "name"))
    )
)
def test_dict_to_sequence_if_pass(value, expected):
    """Ensures that convert our argument to an item"""
    assert dict_to_sequence(value) == expected
```

```
new *
63 @pytest.mark.parametrize(
64     "value, expected",
65     (
66         ({ "Bruins": "Boston",
67            "Blackhawks": "Chicago",
68            "Red Wings": "Detroit",
69            "Canadiens": "Montreal",
70            "Rangers": "New York",
71            "Maple Leafs": "Toronto"},
72          {
73              'Blackhawks': 'Chicago',
74              'Bruins': 'Boston',
75              'Canadiens': 'Montreal',
76              'Maple Leafs': 'Toronto',
77              'Red Wings': 'Detroit',
78              'Rangers': 'New York'
79          }
80     )
81 )
82 def test_from_key_val_list(value, expected):
83     """Ensures that we return an ordered dict"""
84     assert from_key_val_list(value) == expected
85
```

Coverage report: 85%

filter...

coverage.py v7.1.0, created at 2023-02-06 10:16 -0500

Module	statements	missing	excluded	branches	partial	coverage
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\jb_pytest_runner.py	33	3	0	17	7	88%
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\jb_runner_tools.py	173	48	0	50	10	78%
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\jb_serial_tree_manager.py	56	7	0	24	7	88%
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\jb_utils.py	71	34	0	18	0	44%
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\teamcity__init__.py	6	0	0	0	0	100%
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\teamcity\common.py	98	69	0	43	3	33%
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\teamcity\diff_tools.py	62	36	0	17	2	35%
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\teamcity\jb_local_exc_store.py	11	3	0	0	0	73%
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\teamcity\messages.py	153	58	0	30	9	66%
C:\Program Files\JetBrains\PyCharm 2022.3.2\plugins\python\helpers\pycharm\teamcity\pytest_plugin.py	294	123	0	124	32	54%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests__init__.py	68	24	0	10	4	62%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests_version__.py	10	0	0	0	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests_internal_utils.py	19	0	0	2	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\adapters.py	216	14	0	78	7	93%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\api.py	19	3	0	0	0	84%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\auth.py	173	21	0	60	15	84%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\certs.py	4	1	0	2	1	67%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\compat.py	30	2	0	2	1	91%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\cookies.py	239	53	0	98	9	72%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\exceptions.py	37	0	0	2	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\help.py	62	19	0	10	5	66%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\hooks.py	14	0	0	10	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\models.py	455	33	0	188	19	92%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\packages.py	17	0	0	10	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\sessions.py	268	12	0	100	5	95%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\status_codes.py	14	0	0	10	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\structures.py	39	0	0	8	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\utils.py	482	62	0	214	14	87%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests__init__.py	6	2	0	0	0	67%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\compat.py	12	2	0	0	0	83%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\conftest.py	37	3	0	0	0	92%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_help.py	13	0	0	0	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_hooks.py	9	0	0	2	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_lowlevel.py	227	0	0	8	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_packages.py	7	0	0	0	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_requests.py	1651	41	0	78	6	97%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_structures.py	42	0	0	0	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_testserver.py	107	14	0	2	0	87%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\test_utils.py	314	5	0	18	4	97%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\testserver__init__.py	0	0	0	0	0	100%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\testserver\server.py	86	4	0	18	2	94%
C:\Users\youne\OneDrive\Documents\GitHub\requests\tests\utils.py	13	1	0	4	1	88%
Total	5647	688	0	1265	163	85%

coverage.py v7.1.0, created at 2023-02-06 10:16 -0500

1.7 Question 7

Veillez discuter s'il y a des lignes qui sont couvertes par un outil mais pas par l'autre (entre coverage.py et pytest). Quels sont les types de lignes couvertes par l'un mais pas par l'autre ? (Nommez-en 3). Expliquez pourquoi ces lignes sont couvertes/ignorées par ces outils ?

Dans les questions 4 et 5, on remarque la différence entre la couverture de coverage.py, de 91%, et celle de pytest, de 68%.

Pytest n'exécute pas les tests qui n'ont pas la forme « test_*.py ». Cela s'explique par le fait que la commande utilisée précise que l'on veut seulement l'exécution des tests qui ont ce format. Cela explique donc pourquoi on voit dans les exemples ci-dessous que la couverture de code de ces fichiers avec pytest est à 0% alors qu'elle est beaucoup plus haute avec coverage.py.

Pytest :

requests\certs.py	4	4	0	2	0	0%
requests\compat.py	30	30	0	2	0	0%

Coverage.py :

C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\certs.py	4	1	0	2	1	67%
C:\Users\youne\OneDrive\Documents\GitHub\requests\requests\compat.py	30	2	0	2	1	91%