

Класс	Предок	Модуль	Назначение
ChEMBLTargetPipeline	ChEMBLCommonPipeline 1	src/bioetl/pipelines/chembl/ target/run.py	Основная сущность (включая UniProt/TrEMBL) общий класс определяющий pipeline для target

Класс	Предок	Модуль	Назначение
<b>ChemblCommonPipeline</b>	ChemblPipelineBase <sup>8</sup>	<code>src/bioetl/pipelines/chembl/ common/base.py</code>	Общий базовый класс для современных пайплайнов. Обеспечивает проверку входных данных, создание записей и стандартные операции.

Класс	Предок	Модуль	Назначение
<b>ChEMBLPipelineBase</b>	PipelineBase <sup>25</sup>	<code>src/bioetl/core/pipeline/unified.py</code>	Базовый класс для выполнения запросов к ChEMBL, ChemblExp и ChemblEx. Выполнение запросов из API.

Класс	Предок	Модуль	Назначение
<b>ChemblExtractionServiceDescriptor</b>	Generic[ChemblPipelineT] (типовой класс) <sup>38</sup>	<code>src/bioetl/core/pipeline/unified.py</code>	Дескриптор параметров извлечения сущности, использующий подход для формирования запроса, данных и результата. Используются <i>ServiceExtractor</i> для определения данных.
<b>ChemblExtractionDescriptor</b>	object (dataclass) <sup>42</sup>	<code>src/bioetl/pipelines/chembl/common/descriptor.py</code>	Облегчённый извлечение, использующий стратегии идентификации параметров ( <code>mode</code> , <code>batch_size</code> и <code>all</code> ) и планировщик батчи <sup>42</sup> для конвейера <code>dataclass</code> (например).
<b>ServiceExtractionStrategy</b>	—	<code>src/bioetl/pipelines/chembl/common/strategies.py</code>	Стратегия <i>extract</i> с использованием сервисных функций. Поддерживает "service" извлечение <code>run_descriptor</code> у пайплайна дескриптора.

Класс	Предок	Модуль	Назначение
<b>DataclassExtractionStrategy</b>	—	<code>src/bioetl/pipelines/chembl/ common/strategies.py</code>	Стратегия извлечения данных из Chembl через <code>DataclassExtractionStrategy</code> . Поддерживает извлечение данных из кастомных источников, определенных в <code>dataclass_extraction_strategy</code> (например, <code>dataclass_extraction_strategy</code> ).
<b>ExtractionStrategyFactory</b>	—	<code>src/bioetl/pipelines/chembl/ common/strategies.py</code>	Фабрика для создания объектов <code>DataclassExtractionStrategy</code> извлечения данных. Инициализирует набор стратегий по умолчанию и Service, возвращает объект <code>DataclassExtractionStrategy</code> поддержки извлечения данных. Тип дескриптора условий.

Класс	Предок	Модуль	Назначение
ChemblExtractionService	—	src/bioetl/pipelines/chembl/ common/ chembl_extraction_service.py	Сервис, реализующий общие шаги извлечения данных из контекста нормализованного выбора по (клиентскому) финализу. Хранит ChEMBL и определяет ChEMBL. Извлечение execute

Класс	Предок	Модуль	Назначение
<b>ChemblDescriptorFactory</b>	dataclass	<code>src/bioetl/clients/chembl/descriptor_factory.py</code>	Фабрика дескрипторов. Инкапсулирует <i>ChemblConnector</i> и предоставляет метод <code>build_descriptor</code> для подключения к базе данных и построения дескриптора. Метод <code>build_descriptor</code> реализован в <i>ChemblDescriptorFactoryBuilder</i> в файле <code>src/bioetl/clients/chembl/descriptor_factory_builder.py</code> . Метод <code>build_descriptor</code> принимает конфигурацию и возвращает экземпляр <i>ChemblDescriptorFactory</i> или <i>ChemblDescriptorFactoryBuilder</i> в зависимости от указанного параметра.
<b>TargetSchema</b>	<code>pa.DataFrameSchema</code> (Pandera) <sup>78</sup>	<code>src/bioetl/core/schemas/target_schema.py</code>	Pandera-схема для структуры данных <code>target</code> . Обязательные поля: <code>target_id</code> , <code>organism</code> , <code>business_id</code> , <code>business_name</code> , <code>row_has_target</code> . Условия: <code>target_id</code> должен быть не пустым, <code>organism</code> должен быть одним из значений <code>78</code> , <code>79</code> , <code>80</code> . Проверка: данные должны быть уникальными по <code>target_id</code> .
<b>DefaultValidationService</b>	—	<code>src/bioetl/core/pipeline/services/defaults.py</code>	Сервис валидации по схеме. Принимает объект <code>DataFrame</code> и обеспечивает валидацию этой схемы. Сортирует результаты по дате и времени. Используются реализации <code>DefaultValidationService</code> и <code>DefaultValidationServiceBuilder</code> .

Класс	Предок	Модуль	Назначение
<b>ChemblWriteService</b>	—	src/bioetl/pipelines/chembl/ common/base.py	Сервис записи данных в ChEMBL-базу. Принимает на вход детерминированный набор именований, формирует CSV-файл с названиями соединений, качествами, метаданными, затем сохраняет в CSV и записывает в базу. В режиме тестирования вызывает запись метаданных в пайплайн.
<b>PipelineOutputService</b>	—	src/bioetl/core/io/ output_service.py	Сервис-утилита для работы с результатами. Использует унифицированную конфигурацию, пытается записать настроенные UnifiedOutput-конфигурации (resolve) в заданную директорию. DataFrame-объекты преобразует в артефакты, бросает в очередь настроек пайплайна. ChemblTask использует атомарные операции.



Класс	Предок	Модуль	Назначение
<b>WriteArtifacts</b>	dataclass	<code>src/bioetl/core/io/artifacts.py</code>	Структура хранения файлов по полям: <code>data</code> (основной <code>data_path</code> данных), <code>YAML-файл манифеста</code> ( <code>manifest</code> ), <code>quality</code> CSV файл ( <code>qc_summary</code> ) сводки <code>QC</code> для доп. артефактов, заполняемых сохранением (например, формируемые
<b>ChEMBLClient</b>	ConfiguredHttpClient <sup>95</sup>	<code>src/bioetl/clients/chembl/client.py</code>	Клиент для унифицированной архитектуры базовый ( <i>Configure</i> ) обеспечить запросов (например, с поддержкой фильтров инициализации конфигурации ChEMBL и <i>HttpBackends</i> Requests) выполнение

Класс	Предок	Модуль	Назначение
<b>ChemblRequestBuilder</b>	—	<code>src/bioetl/clients/chembl/ client.py</code>	Вспомогательный класс для формирования запросов к ChEMBL API. Является частью <code>dataclasses</code> и используется для конфигурирования маршрута запроса. Объект <code>ChemblRequestBuilder</code> выполняется в методе <code>101</code> <code>ChemblClient</code> для формирования запросов к ресурсам ChEMBL.
<b>UniProtClient</b>	<code>ConfiguredHttpClient</code> <sup>104</sup>	<code>src/bioetl/clients/uniprot/ client.py</code>	Клиент для взаимодействия с UniProt. Реализован в соответствии с архитектурой наследования. Наследует от <code>ConfiguredHttpClient</code> (аналогично <code>104</code> ). Создан для получения информации из источника <code>HttpBackedClient</code> . Предоставляет методы для формирования запросов и получения данных.



Класс	Предок	Модуль	Назначение
<b>TargetNormalizer</b>	—	<code>src/bioetl/pipelines/chembl/target/normalizers.py</code>	Класс-шунормализации основан на словарях. Предполагается, что и стандартные (например, извлечение) и нестандартные (например, извлечение) данные, однако те, которые не заполнены, будут обработаны. NotImplementedError

1 2 3 4 5 6 7	<b>run.py</b>	<a href="https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/chembl/target/run.py">https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/chembl/target/run.py</a>
8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41	<b>base.py</b>	<a href="https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/chembl/common/base.py">https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/chembl/common/base.py</a>
42 43	<b>descriptor.py</b>	<a href="https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/chembl/common/descriptor.py">https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/chembl/common/descriptor.py</a>
44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59	<b>strategies.py</b>	<a href="https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/chembl/common/strategies.py">https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/chembl/common/strategies.py</a>
60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75	<b>chembl_extraction_service.py</b>	<a href="https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/chembl/common/chembl_extraction_service.py">https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/chembl/common/chembl_extraction_service.py</a>
76 77	<b>descriptor_factory.py</b>	<a href="https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/clients/chembl/descriptor_factory.py">https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/clients/chembl/descriptor_factory.py</a>
78 79	<b>target_schema.py</b>	<a href="https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/core/schemas/target_schema.py">https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/core/schemas/target_schema.py</a>
80 81 82 83 84	<b>defaults.py</b>	<a href="https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/core/pipeline/services/defaults.py">https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/core/pipeline/services/defaults.py</a>
89 90 91 92 93	<b>output_service.py</b>	<a href="https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/core/io/output_service.py">https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/core/io/output_service.py</a>

94 **artifacts.py**

[https://github.com/SatoryKono/bioactivity\\_data\\_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/core/io/artifacts.py](https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/core/io/artifacts.py)

95 96 97 98 99 100 101 102 103 **client.py**

[https://github.com/SatoryKono/bioactivity\\_data\\_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/clients/chembl/client.py](https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/clients/chembl/client.py)

104 105 106 107 **client.py**

[https://github.com/SatoryKono/bioactivity\\_data\\_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/clients/uniprot/client.py](https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/clients/uniprot/client.py)

108 109 110 111 112 113 114 115 116 117 118 **factories.py**

[https://github.com/SatoryKono/bioactivity\\_data\\_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/clients/chembl/factories.py](https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/clients/chembl/factories.py)

119 120 121 122 **client\_registry.py**

[https://github.com/SatoryKono/bioactivity\\_data\\_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/client\\_registry.py](https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/client_registry.py)

123 124 **normalizers.py**

[https://github.com/SatoryKono/bioactivity\\_data\\_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/chembl/target/normalizers.py](https://github.com/SatoryKono/bioactivity_data_acquisition/blob/98f17f432532cddd56d0a07fd4796b37c54677ec/src/bioetl/pipelines/chembl/target/normalizers.py)