ultima-reports-api:1.2.2

В данном разделе приведено описание API подсистемы подготовки отчётов Платформы НЕЙРОСС.

Основные классы API в составе артефакта ultima-reports-api_<версия>.jar:

Класс	Описание
`extensions.reports.ReportDefinition`	Базовый абстрактный класс для всех классов реализации шаблонов отчётов. Основные методы класса: def init(conf: Config, loader: ReportDefinitionLoader): Unit В реализации данного метода следует выполнить инициализацию шаблона. Метод вызывается логикой Платформы НЕЙРОСС в тот момент, когда пользователь хочет сформировать отчёт по выбранному шаблону. В метод передаются объекты классов Config (соответствует содержимому файла report.conf) и ReportDefinitionLoader — последний позволяет получить доступ к файлам в папке шаблона отчёта (например, для загрузки jasper-файла). def title: String Должен возвращать название шаблона, которое будет выведено в веб-интерфейсе Платформы НЕЙРОСС. def description: String Должен возвращать описание шаблона, которое будет выведено в веб-интерфейсе Платформы НЕЙРОСС. def renderForm(...): Result def renderFormAction(...): Action Генерация и выдача формы ввода параметров для формирования отчёта. Два метода позволяют как переопределить работу с Action (при использовании views в Play Framework, например), так и просто обработать «голый» HTTP-запрос и вернуть «голый» HTTP-ответ.

	def processForm(...): AnyRef Обработка введённых пользователем параметров. Метод должен вернуть объект (любого типа), содержащий разобранные параметры. Этот объект впоследствии будет передан в метод generateReport. Если параметры заданы неверно, то метод должен выбросить исключение. def generateReport(format: String, form: AnyRef, dataSource: javax.sql.DataSource, to: File) Данный метод вызывается логикой Платформы НЕЙРОСС для непосредственной генерации отчёта. В параметрах передаются: 1. требуемый формат отчёта (один из заданных в `report.conf`) — например, `"pdf"`. 2. объект параметров, который ранее был получен вызовом метода `processForm`. 3. объект доступа к источнику данных; 4. хэндлер файла, в который следует записать сгенерированный отчёт. def customAction(path: String): Action Данный метод позволяет реализовать кастомное HTTP-API для шаблона отчёта. Например, выдавать в форму отчёта список объектов из базы данных по запросу и пр.
`extensions.reports.JasperReportDefinition`	Класс реализации шаблона отчёта с поддержкой JasperReports Library, который формирует отчёт на основе jasper-шаблонов. Является потомком extensions.reports.ReportDefinition. Можно использовать этот класс как есть, если не требуется поддержки формы ввода параметров.

`extensions.reports.GenericFormField`	Интерфейс для класса поля ввода. Определяет обязательные методы любого класса поля ввода: def key(): String Должен возвращать имя параметра, для которого определено данное поле ввода. def title(): String Должен возвращать удобочитаемое название поля ввода. Данный текст выводится в форме в интерфейс пользователя. def render(): Html Должен возвращать объект-представление HTML-фрагмента поля ввода. def parse(value: String): AnyRef Преобразует введённое строковое значение параметра к Java-объекту требуемого типа.
`extensions.reports.GenericFormSupport`	Утилитарный трэйт поддержки базовой формы ввода параметров. Хранит карту объектов полей ввода (потомков extensions.reports.GenericFormField), генерирует базовую форму ввода параметров со списком полей ввода. Основные методы: def registerField(field: GenericFormField): Unit Регистрирует новое поле ввода. Данный метод должен вызываться в классах реализации шаблонах, наследуемых от extensions.reports.GenericFormSupport, для составления списка полей ввода. def renderFormAction: Action[AnyContent] Генерирует базовую форму ввода параметров на основе заданного через registerField списка полей ввода. def processForm(request: Request [MultipartFormData[TemporaryFile]]): AnyRef Выполняет разбор значений формы ввода параметров, вызывая метод extensions.reports.GenericFormField.parse (...) и формируя карту вида «имя параметра — значение».

`extensions.reports.JasperFormField`	Интерфейс поля ввода для Jasper-параметра. Является потомком extensions.reports.GenericFormField. Определяет новый метод def init (jrParameter: JRParameter): Unit, который вызывается в extensions.reports.GenericJasperReportDefinition при инициализации списка полей ввода по Jasper-параметрам.
`extensions.reports.GenericJasperReportDefinition`	Класс базовой реализации шаблона отчёта, который формирует отчёт на основе jasper-шаблонов и умеет генерировать по jasper-файлу форму для ввода параметров. Является потомком extensions.reports.JasperReportDefinition и extensions.reports.GenericFormSupport.
`extensions.reports.generic.AbstractJasperFormField`	Абстрактный класс базовой реализации поля ввода для Jasper-параметра. Реализует интерфейс extensions.reports.JasperFormField: def key(): String Возвращает имя (name) Jasper-параметра в роли ключа. def title(): String Возвращает описание (description) Jasper-параметра в роли названия поля.
`extensions.reports.generic.DateTimeFormField`	Реализация поля ввода даты / времени. Возможности по использованию данного поля описаны в Поля ввода параметров для отчётов JasperReports.
`extensions.reports.generic.InputFormField`	Реализация полей ввода текста / числа. Возможности по использованию данного поля описаны в Поля ввода параметров для отчётов JasperReports.
`extensions.reports.generic.SQLChoiceFormField`	Реализация поля выбора опций из списка. Опции загружаются из базы данных в соответствии с заданным SQL-запросом. Возможности по использованию данного поля описаны в Поля ввода параметров для отчётов JasperReports.

`extensions.reports.generic.OriginFormField`	Реализация скрытого поля для передачи в отчёт базового URL сервера Платформы НЕЙРОСС. Для встраивания гиперссылок в отчёт. Возможности по использованию данного поля описаны в Поля ввода параметров для отчётов JasperReports.