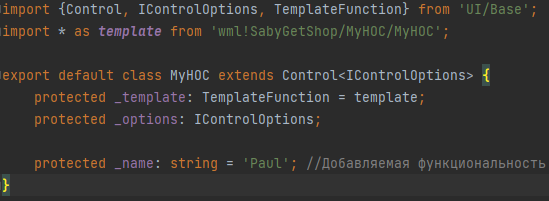
# Wasaby templates

### Пример реализации HOC

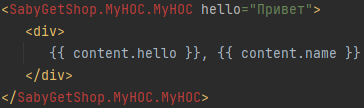
* MyHOC.ts:



* MyHOC.wml:



* Использование MyHOC:



В HOC передали в качестве опции строку «Привет». HOC в свою очередь передает все свои опции в тот контент, который оборачивает, а так же добавляет в опции контента еще и свою переменную.

Результат:



Итого: в оборачиваемый компонент прозрачно переданы все опции + добавленная функциональность. В partial может быть отрендерен передаваемый компонент из контентной опции плюс своя верстка с помощью директивы <ws:template>.

### Директива <ws:template>. Встроенный шаблон

**Только** в таком шаблоне опции доступны по прямому названию аттрибута, без options.

### Классы ws-is…

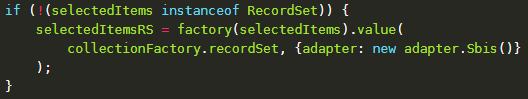
Wasaby автоматически добавляет специальные CSS-классы для тега body. С их помощью вы можете определить какой тип управления используется в настоящий момент. Например, если страница загрузилась на телефоне, то на body будет класс ws-is-phone

# TypeScript

### Из List в RecordSet







### Механизм предзагрузки

Если перейти в компонент спа-переходом, то предзагрузчик отработает на клинете. Если страница загружается по ссылке — предзагрузчик отработает на сервере.

### Хардкод RecordSet

const serverResponse = {

\_type: 'recordset',

d: <Массив с помощью JSON>,

s: <Массив с помощью JSON>,

};

this.\_hardData = new RecordSet({

rawData: serverResponse,

adapter: new adapter.Sbis()

});

### Испольнить метод в зависимости от условия



this[result.id ? '\_reloadItem' : 'reloadList'](result.id || ...[]);

### Серверная отладка

import { IoC } from 'Env/Env';

const logger = IoC.resolve('ILogger');

logger.error('BookingTimeFrame', err);

### Класс-предзагрузчик и функциональный предзагрузчик (getConfig)

В index.ts в аргументах конфига, в свойстве “loaders” и в pagex в “customLoaders” должны быть классы-предзагрузчики. А в pagex в “prefetchConfig“ должен быть указан модуль с функцией getConfig

# Others

### WS

* Alt + Shift + выделение — поставить курсор на много строк

# Fiddler

### Запуск wather’а

npm run serve

Будет билдить автоматом в application при изменении

### ReceivedState

У многих платформенных контролов есть receivedstate, даже если прокидывается мемори или префетч сверху. Поэтому такие изменения могут не применяться. В таком случае можно воткнуть на такой контрол if="{{ \_isMounted или \_mounted }}" в шаблоне, естественно потом убрать. Это заставит его перерисоваться с новыми опциями при оживлении