

概要

- 概要
- 編集ダイアログの実装
 - ダイアログを実装する
 - 編集動作の実装
 - 確定ボタン動作の実装

編集ダイアログの実装

ダイアログを実装する

以下を参考にフォームを内包したダイアログを作成する

<https://element.eleme.io/#/en-US/component/dialog#customizations>

```
<el-dialog title="Todo 編集" :visible.sync="dialogFormVisible">
  <el-form :model="form">
    <el-form-item label="ID" :label-width="formLabelWidth">
      <el-input v-model="form.id"></el-input>
    </el-form-item>
    <el-form-item label="タイトル" :label-width="formLabelWidth">
      <el-input v-model="form.title"></el-input>
    </el-form-item>
    <el-form-item label="内容" :label-width="formLabelWidth">
      <el-input v-model="form.content"></el-input>
    </el-form-item>
    <el-form-item label="状態" :label-width="formLabelWidth">
      <el-select v-model="form.status" placeholder="選択してください">
        <el-option label="TODO" value="TODO"></el-option>
        <el-option label="PROGRESS" value="TODO"></el-option>
        <el-option label="DONE" value="DONE"></el-option>
      </el-select>
    </el-form-item>
  </el-form>
  <span slot="footer" class="dialog-footer">
    <el-button @click="dialogFormVisible = false">中止</el-button>
    <el-button type="primary" @click="dialogFormVisible = false">確定</el-button>
  </span>
</el-dialog>
```

ダイアログ制御用の変数とフォーム用の変数を追加します。

```
formLabelWidth: '150px',
dialogFormVisible: false,
disableID: true,
rowNumber: '',
form: {
```

```
id: null,  
title: '',  
content: '',  
status: ''  
}
```

編集ボタンクリック時にダイアログが表示されるようにします。

```
handleEdit(index, row) {  
  console.log(index, row)  
  this.dialogFormVisible = true  
},
```

編集動作の実装

編集ボタンをクリックされた時に、該当する行の内容をダイアログに表示する動きを実装します。(ただし、バックエンドとつないだ時には、バックエンドにデータ取得しに行くように変更します。)

- IDは入力できないように「disableID」をtrueにします。
- クリックされた行「row」の内容を「form」にコピーします。

```
handleEdit(index, row) {  
  this.disableID = true  
  this.rowNumber = index  
  // ToDo: 本来はREST APIのkey(row.id)検索し結果をformにセットする  
  this.form.id = row.id  
  this.form.title = row.title  
  this.form.content = row.content  
  this.form.status = row.status  
  this.dialogFormVisible = true  
},
```

確定ボタン動作の実装

確定ボタンをクリックしたときの動作をダミーで実装します。
バックエンドとつないだ時には、更新のAPIをコールするように変更します。

methodsにdoConfirmを追加します。

- ダイアログを閉じる(非表示にする)
- メッセージ用にtargetを作成
- 編集結果を対象の行に反映する
- 処理成功のメッセージを表示する

```
doConfirm() {  
  this.dialogFormVisible = false
```

```
const target = `${this.form.id}:${this.form.title}`
// ToDo: REST APIのアップデートを起動する
const row = this.$refs.itemTable.data[this.rowNumber]
row.title = this.form.title
row.content = this.form.content
row.status = this.form.status
this.$message({
  type: 'success',
  message: `${target} : 更新が成功しました。`,
  showClose: true,
  duration: 5000
})
}
```

確定ボタンクリック時にdoConfirmをコールするよう変更します。

```
<el-button type="primary" @click="doConfirm()">
```