

# ( 디지털 컨버전스 ) 스마트 콘텐츠와 웹 융합 응용 SW개발자 양성과정

훈련기간 : 2021.05.07 ~ 2021.12.08

오목 props&emit 복합

```
<template>
  <div>
    <div>{{ turn }}님의 턴입니다.</div>
    <table-component
      :table-data="tableData"
      :turn="turn"
      :winner="winner"
      v-on:updateTurn="updateTurn"
      v-on:updateTableData="updateTableData"
      v-on:updateWinner="updateWinner">
    </table-component>
    <div v-if="winner">{{ winner }}님의 승리!</div>
  </div>
</template>
<script>
import TableComponent from '@components/concave/TableComponent.vue'
export default {
  components: {
    TableComponent
  },
  data () {
    return {
      turn: 'O',
      winner: '',
      tableData: [
        ['', '', ''],
        ['', '', ''],
        ['', '', '']
      ]
    }
  },
  methods: {
    updateTurn (val) {
      this.turn = val
    },
    updateTableData () {
      this.tableData = [
        ['', '', ''],
        ['', '', ''],
        ['', '', '']
      ]
    },
    updateWinner (val) {
      this.winner = val
    }
  }
}
</script>
```

```
TableComponent.vue
<template>
  <table>
    <tr-component
      v-for="(rowData, index) in tableData"
      :key="index"
      :row-data="rowData"
      :row-index="index"
      :table-data="tableData"
      :turn="turn"
      :winner="winner"
      v-on:updateTurn="updateTurn"
      v-on:updateTableData="updateTableData"
      v-on:updateWinner="updateWinner"
      v-model="propTurn">
    </tr-component>
  </table>
</template>
```

```
TrComponent.vue
<template>
  <tr>
    <td-component
      v-for="(cellData, index) in rowData"
      :key="index"
      :cell-data="cellData"
      :cell-index="index"
      :row-index="rowIndex"
      :table-data="tableData"
      :turn="turn"
      :winner="winner"
      v-on:updateTurn="updateTurn"
      v-on:updateTableData="updateTableData"
      v-on:updateWinner="updateWinner"
      v-model="propTurn">
  </tr>
</template>
```

```
TdComponent.vue
<template>
  <td @click="onClickTd">{{ cellData }}</td>
</template>
props: {
  cellData: String,
  rowIndex: Number,
  cellIndex: Number,
  tableData: Array,
  turn: String,
  winner: String
},
```

```
onClickTd () {
  if (this.cellData) {
    return
  }
  this.$set(this.tableData[this.rowIndex], this.cellIndex, this.turn)
  let win = false

  if (
    this.tableData[this.rowIndex][0] === this.turn &&
    this.tableData[this.rowIndex][1] === this.turn &&
    this.tableData[this.rowIndex][2] === this.turn
  ) {
    win = true
  } else if (
    this.tableData[0][this.cellIndex] === this.turn &&
    this.tableData[1][this.cellIndex] === this.turn &&
    this.tableData[2][this.cellIndex] === this.turn
  ) {
    win = true
  } else if (
    this.tableData[0][0] === this.turn &&
    this.tableData[0][1] === this.turn &&
    this.tableData[0][2] === this.turn
  ) {
    win = true
  } else if (
    this.tableData[1][0] === this.turn &&
    this.tableData[1][1] === this.turn &&
    this.tableData[1][2] === this.turn
  ) {
    win = true
  } else if (
    this.tableData[2][0] === this.turn &&
    this.tableData[2][1] === this.turn &&
    this.tableData[2][2] === this.turn
  ) {
    win = true
  }

  if (win) {
    this.$emit('updateWinner', this.winner)
    this.$emit('updateTurn', 'O')
    this.$emit('updateTableData')
  } else {
    let all = true
    this.tableData.forEach(row => {
      row.forEach(cell => {
        if (!cell) {
          all = false
        }
      })
    })
    if (all) {
      this.propWin = ''
      this.$emit('updateTurn', 'O')
      this.$emit('updateTableData')
    } else {
      this.game.propTurn = this.turn === 'O' ? 'X' : 'O'
      this.$emit('updateTurn', this.game.propTurn)
    }
  }
}
```

이긴경우 해당턴에 이긴사람이 승자  
승자 설정  
승자 부모 컴포넌트로 전달  
시작 O의 턴으로 리셋  
각 행의 열을 찾아서 비어있다면 all을 false로 둬  
true가 나온다면 테이블을 새로 리셋한다.  
false의 경우 턴 교체

출력

O		O
	X	X
		O

X님의 턴입니다

X님의 승리!

테이블

```
<tr>
  <td-component
    v-for="(cellData, index) in rowData"
    :key="index"
    :cellData="cellData"
    :row-index="index"
    :table-data="tableData"
    :turn="turn"
    :winner="winner"
    v-on:click="updateTurn"
    v-on:click="updateTableData"
    v-on:click="updateWinner"
    v-model="propTurn"
  </td-component>
</tr>
```

```
<table>
  <tr-component
    v-for="(rowData, index) in tableData"
    :key="index"
    :row-data="rowData"
    :row-index="index"
    :table-data="tableData"
    :turn="turn"
    :winner="winner"
    v-on:click="updateTurn"
    v-on:click="updateTableData"
    v-on:click="updateWinner"
    v-model="propTurn"
  </tr-component>
</table>
```

```
table {
  border-collapse: collapse;
}
td {
  border: 1px solid black;
  width: 80px;
  height: 80px;
  text-align: center;
}
```

```
<table> == $0
  <tr value="0">
    <td value="0">O</td>
    <td value="0">X</td>
    <td value="0">O</td>
  </tr>
  <tr value="0">...</tr>
  <tr value="0">...</tr>
</table>
```