1. 공통 적용 style 인 common.css 구성

* 하기 이슈 포함하여 필요한 class 및 기타 요소 공통으로 대응 되도록 작성함.

1. Input 에 boz-sizing:border-box 적용으로 데이터 값이 가려지는 이슈.

* Width 를 4~8px 가량 일괄적으로 넓게 작업하여 대응 하기로 함.

1. Input 에 사용자가 값을 입력 시, input 배경색이 브라우저에 의해 변경되는 이슈 대응
2. 입력항목의 입력값의 사이즈가 커지는 경우

- 글자 크기 및 입력창 높이 고정 (height:20px; line-height:20px; font-size:12px;) 으로 대응함.

5. 필수 입력 부분의 input 의 배경색이 브라우저에서 지정한 배경색으로 변경 됨

- 필수 입력 부분의 input 태그에 적용한 배경색이 자동완성이 켜져 있고,

그 값을 선택 할 경우 브라우저에서 지정한 배경색으로 변경되는 문제를

하기 css를 사용하여 대응함.

input:-webkit-autofill,

input:-webkit-autofill:hover,

input:-webkit-autofill:focus,

input:-webkit-autofill:active {

transition: background-color 5000s ease-in-out 0s;

-webkit-transition: background-color 9999s ease-out;

-webkit-box-shadow: 0 0 0px 1000px white(지정하고자 하는 배경색 및 배경색코드) inset !important;

-webkit-text-fill-color: #fff(지정하고자 하는 글자색 및 글자색코드) !important;

}

input:-ms-autofill,

input:-ms-autofill:hover,

input:-ms-autofill:focus,

input:-ms-autofill:active {

transition: background-color 5000s ease-in-out 0s;

-ms-transition: background-color 9999s ease-out;

-ms-box-shadow: 0 0 0px 1000px white(지정하고자 하는 배경색 및 배경색코드) inset !important;

-ms-text-fill-color: #fff(지정하고자 하는 글자색 및 글자색코드) !important;

}

6. 왼쪽 메뉴 부분 사이즈 및 스크롤 관련 개선작업

- leftMenu.jsp 중 makeTree(), makeSpan() 내부의 css 값을 조정하여, 컨탠츠 크기(메뉴 span의 width)에 따라 자동으로 x축의 스크롤바가 생성 되도록 개선함. 개선 내용은 하기에 기재함.

* makeSpan() 개선 내용

obj.style.width = "400px"; → obj.style.width = "fit-content";

* makeTree() 추가 내용

subMenu.style.minWidth = 184+'px';  
subMenu.style.paddingRight = 4+'px';

7. 거래 컨텐츠의 양쪽 맞춤 정렬 가능 여부에 대한 검토

- 각 행을 div 및 P 태그로 묶은 요소의 경우 여러가지로 양쪽 정렬에 가장 적합함.

- table 구조로 묶여 있는 경우 display:flex 속성을 이용하거나, 정렬 방법을 사용하면 정리가 가능한 수준이나, 원하는 결과를 얻기 위해서는 세부적인 조정이 필요함.

- 각 input 및 label 로만 구성 된 부분은 layout이 잡히지 않아, DOM 구조의 재정립이 필요할 것으로 확인 됨.

8. 양쪽 맞춤 정렬에 필요한 조치 정리

- table 을 이용한 table 코딩의 연장으로 현재 각 label 과 input 으로 개별화 되어 있는 요소들을 table에 위치시켜 최소한의 규격을 잡는 방법으로 세부적인 정렬 및 수치를 적용하는데 불편함이 있을 수 있으며 layout 표현에도 한계가 존재함.

- br로 구분되어 각 tag(labrl, input)별로 위치를 잡았던 요소들을 div 등으로 묶어서 행으로 정리하는 방법으로 table 방식에 비해 정렬 및 수치 적용, layout 표현에도 한계가 줄어들 수 있음.

- 서로 짝지어 있는 label 및 input 을 하나의 단위로 묶어 grid 및 flex 로 layout을 구성하는 방법으로 각 grid 및 flex 로도 표현이 힘든 부분들에 대한 세부적인 조정이 필요하며, 각 거래화면에 대해 flex 및 grid에 맞춘 css 적인 설계가 동반되어야 함. 현실적으로 상단 항목인 각 행으로 감싸는 div 작업이 동반 될 가능성이 높음.

- 각 간격을 px이 아닌 min-width 기준의 % 단위로 바꾸는 방법도 사용 가능하나, 현재 label의 width만 사용하여 간격을 잡은 곳이 상당히 많기 때문에 이를 각 label의 실 width 값으로 전부 변환하는 작업도 같이 진행 되어야 함. 작업과 별개로 무한정 데이터 간의 간격이 벌어질 경우, 한번에 거래 화면이 다 보이지 않을 수 있고 layout이 어딘가 비어 보이고 어색해 보일 수 있음. 또는 단순히 여백이 늘어나는 것만이 아닌 입력을 위한 input이 늘어나기를 바라는 요구도 있을 수 있음(이는 상기 항목에서도 동일하게 나타날 상황이므로 고객 니즈 파악 및 내부 논의가 필요함).

- 해당 부분에 대한 시각화를 위해 샘플 파일 제작 중.

- 거래번호 12010\_ex.html, 17010\_ex.html 샘플 파일 제작 및 해당 부분에 대해 david 과 논의.

- 상기 양쪽 정렬 샘플 제작 시 common.css 에 해당 layout 구성을 위한 class 지정 및 세팅 작업을 하기와 같이 진행.

**< -- html 작성 예 -- >**

**<div class="wrap\_row">**

**<div class="row\_50">**

**<label class="tab\_page\_label" style='margin-left:36px'>元金返済方式</label>**

**<select name='f77' table='LNPCODE' key='507' disabled=true></select> </div>**

**<div class="row\_50">**

**<label class="tab\_page\_label" style='margin-left:-23px'>利息返済方式</label>**

**<select name='f78' table='LNPCODE' key='506' disabled=true></select><br>**

**</div>**

**</div>**

**< -- css 작성 예 -- >**

**/\* ---- 양쪽 맞춤 정렬 사용 시 각 행 및 구성 요소를 묶어 배치 하는데 사용 ---- \*/**

**/\* -- 행으로 요소를 묶는데 사용 -- \*/**

**.wrap\_row{position:relative; display:block; box-sizing:border-box; background:transparent; width:100%; border:0; padding:0; margin:0; font-size:0;}**

**/\* -- --------------------- -- \*/**

**/\* -- 짝지어 있는 label 및 input 을 행 안에 한 단위로 묶을 때 사용 -- \*/**

**.wrap\_row:after{content:""; position:relative; display:block; width:100%; height:0; clear:both;}**

**.wrap\_row .row\_05{position:relative; display:inline-block; box-sizing:border-box; width:calc(5% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_10{position:relative; display:inline-block; box-sizing:border-box; width:calc(10% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_15{position:relative; display:inline-block; box-sizing:border-box; width:calc(15% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_20{position:relative; display:inline-block; box-sizing:border-box; width:calc(20% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_25{position:relative; display:inline-block; box-sizing:border-box; width:calc(25% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_30{position:relative; display:inline-block; box-sizing:border-box; width:calc(30% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_33{position:relative; display:inline-block; box-sizing:border-box; width:calc(33% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_35{position:relative; display:inline-block; box-sizing:border-box; width:calc(35% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_40{position:relative; display:inline-block; box-sizing:border-box; width:calc(40% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_45{position:relative; display:inline-block; box-sizing:border-box; width:calc(45% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_50{position:relative; display:inline-block; box-sizing:border-box; width:calc(50% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_55{position:relative; display:inline-block; box-sizing:border-box; width:calc(55% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_60{position:relative; display:inline-block; box-sizing:border-box; width:calc(60% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_65{position:relative; display:inline-block; box-sizing:border-box; width:calc(65% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_66{position:relative; display:inline-block; box-sizing:border-box; width:calc(66% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_70{position:relative; display:inline-block; box-sizing:border-box; width:calc(70% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_75{position:relative; display:inline-block; box-sizing:border-box; width:calc(75% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_80{position:relative; display:inline-block; box-sizing:border-box; width:calc(80% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_85{position:relative; display:inline-block; box-sizing:border-box; width:calc(85% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_90{position:relative; display:inline-block; box-sizing:border-box; width:calc(90% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_95{position:relative; display:inline-block; box-sizing:border-box; width:calc(95% - 0.1px); font-size:12px;}**

**.wrap\_row .row\_100{position:relative; display:block; box-sizing:border-box; width:100%; font-size:12px;}**

**/\* -- --------------------------------------------------- -- \*/**

**/\* ---- --------------------------------------------------------- ---- \*/**

**/\* ---- 필요 시 독자적으로 사용 가능, 상단 양쪽 맞춤 정렬 시에도 혼합하여 사용 ---- \*/**

**.white\_nowrap{white-space:nowrap!important;}**

**.text\_left{text-align:left!important;}**

**.text\_center{text-align:center!important;}**

**.text\_right{text-align:right!important;}**

**/\* ---- ------------------------------------------------------------ ---- \*/**

* 기존 table 구조 에서 3단 이상으로 구현 된 구조에 현재 구조를 같이 쓸 수 있는지 layout 을 시험 한 결과 기존 table의 html 구조를 일부 변경 해야 하며, 기타 css 적인 처리 및 속성 변경도 다수 해줘야 하는 점을 확인함.
* 거래내역 화면 전체에 padding-left 를 20px 을 주는 .dataform\_content(거래 화면 최상단 name="dataForm"에 적용) 를 각 거래 파일 마다 추가 하기로 함 (luke 검토 요처 - David, Bella 와 논의 후 적용) - 현재 lupin 이 부트스트랩 적용을 확인 중으로 확인이 끝난 후 적용 시키기로 함.