



융합프로그래밍실습2_3주

2020088359 김경현, 2022045332 김시온,
2022022488 박준호, 2022088913 이재욱





목 차

Contents

I. 프로젝트 소개

II. ERP System 디자인

- 시작 Dialog & 주문 정보 Dialog & 불량 정보 Dialog

III. ERP System 기능

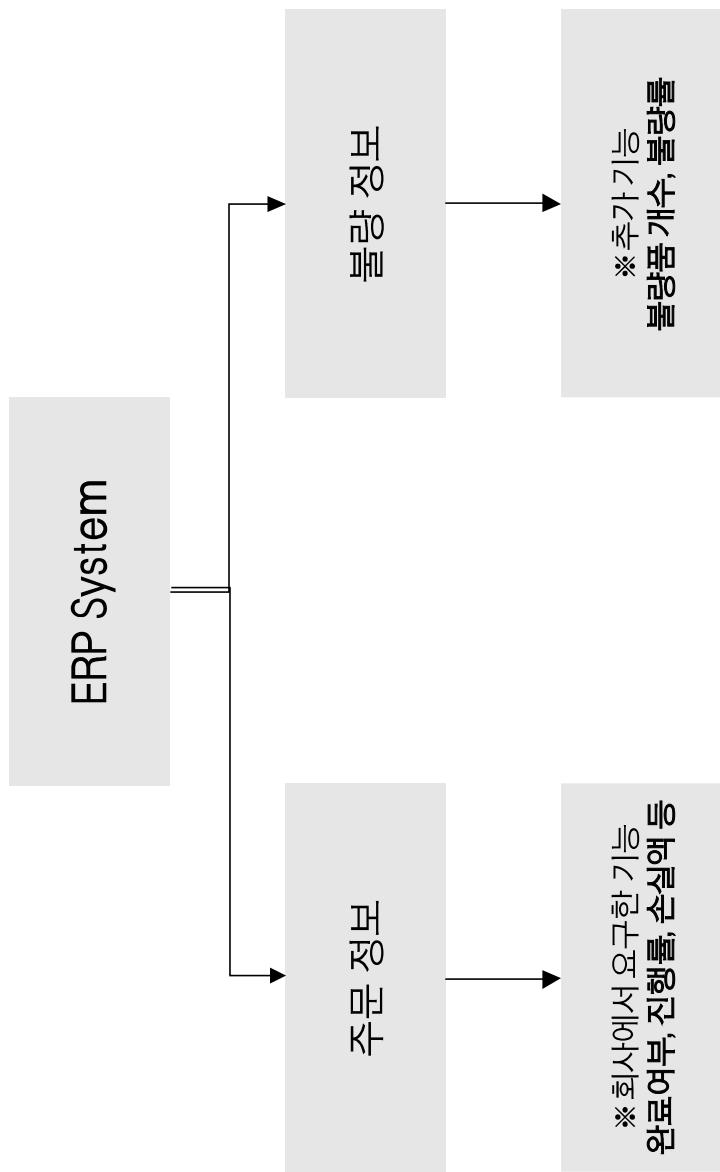
- 시작 Dialog & 주문 정보 Dialog & 불량 정보 Dialog

IV. 코드 구현

V. 동작 영상

프로젝트 소개

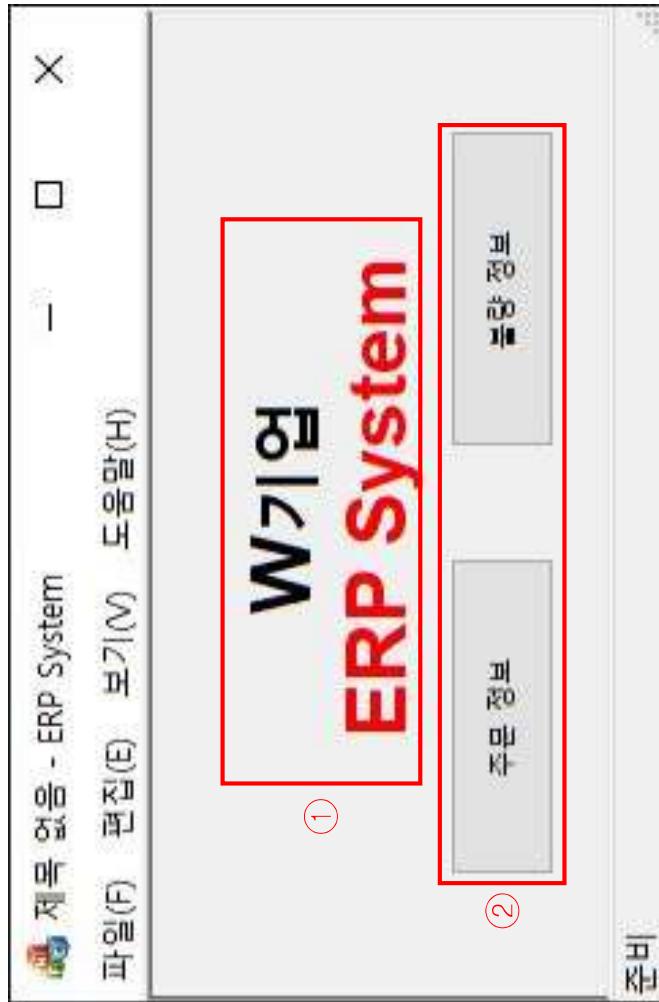
- 우리는 W기업 연구센터에 소속된 개발3팀의 개발자들
- 3주 전 회사로부터 빌주 내역을 관리하기 위한 프로그램 개발을 지시 받음
- 우리는 프로그램 개발을 완료했으며 프로그램의 이름은 “ERP System”



ERP System 디자인 – 시작 Dialog

The Engine of Korea

한양대 ERICA
한양대학교
한양대 ERICA

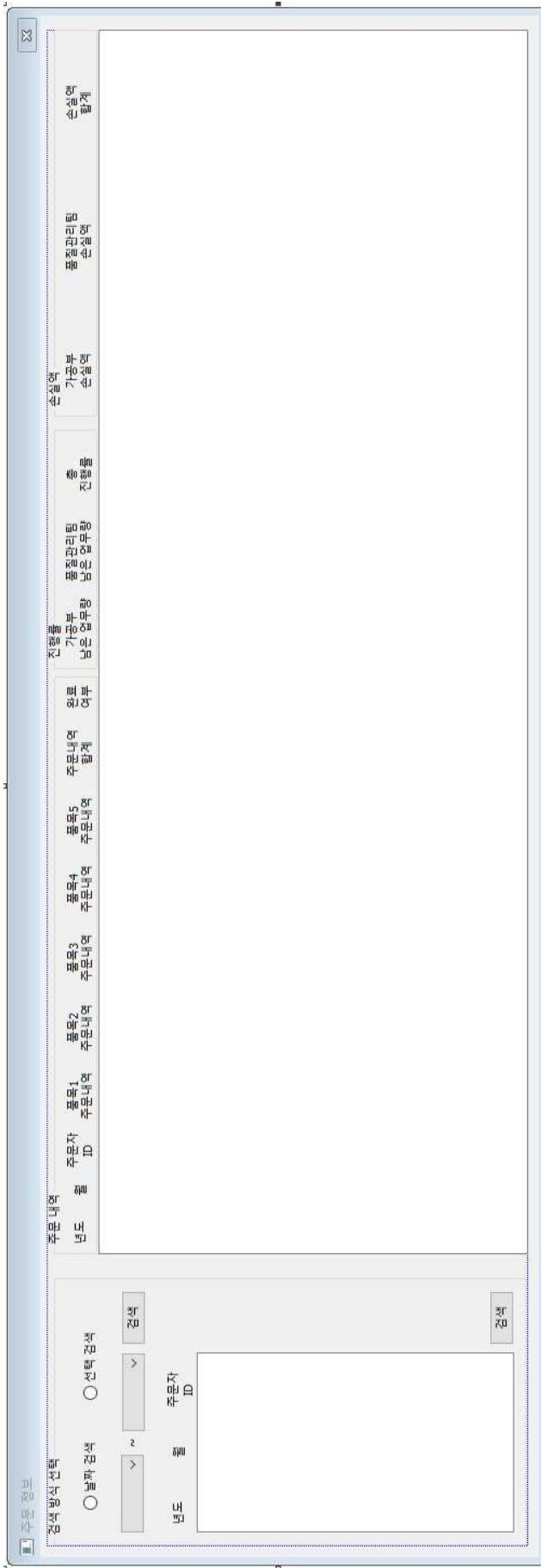


- ① : 프로그램명 (OnDraw)
- ② : 각 정보 Dialog 생성 버튼 (Button)

ERP System 디자인 - 주문 정보 Dialog

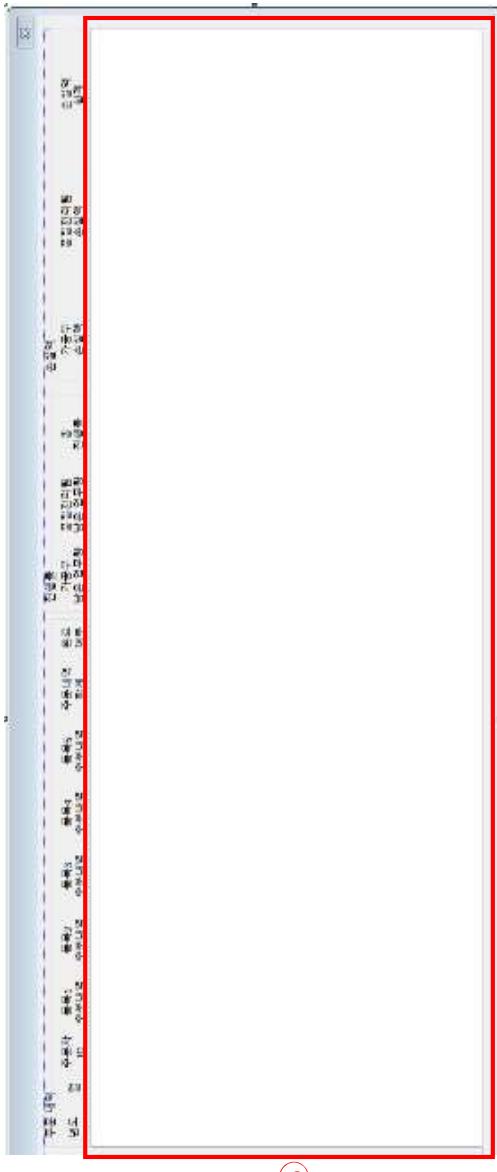
The Engine of Korea
THE KOREAN ECONOMY IN THE 1990S

중간고소통, 참여와 협력의 양 날개를 펼쳐 미래로 날아가는
한양대ERICA



공전합용으로

ERP System 디자인 – 주문 정보 Dialog



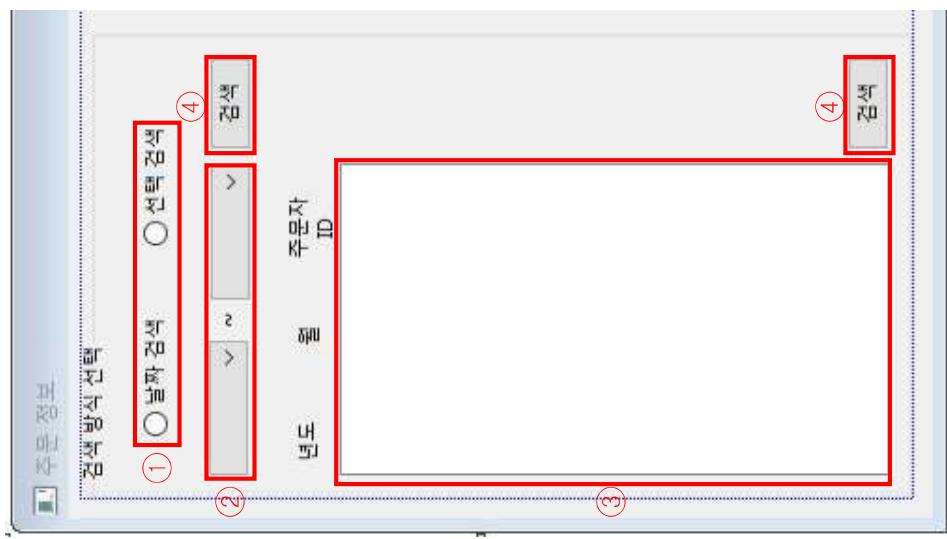
① : 검색 방식 선택 (Radio Button)

② : 날짜 선택 (Combo Box)

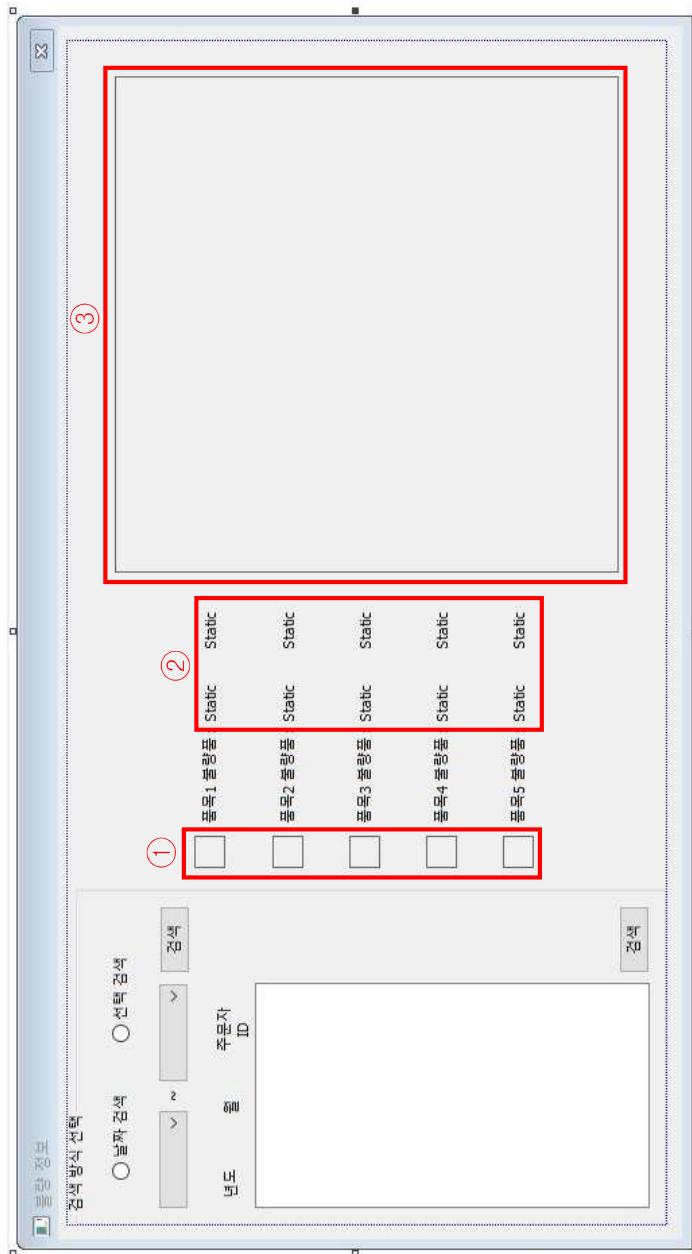
③ : 주문 건수 선택 (List Box : Multiple)

④ : 검색 버튼 (Button)

⑤ : 검색 결과 표시 (List Box : None)



ERP System 디자인 – 불량 정보 Dialog

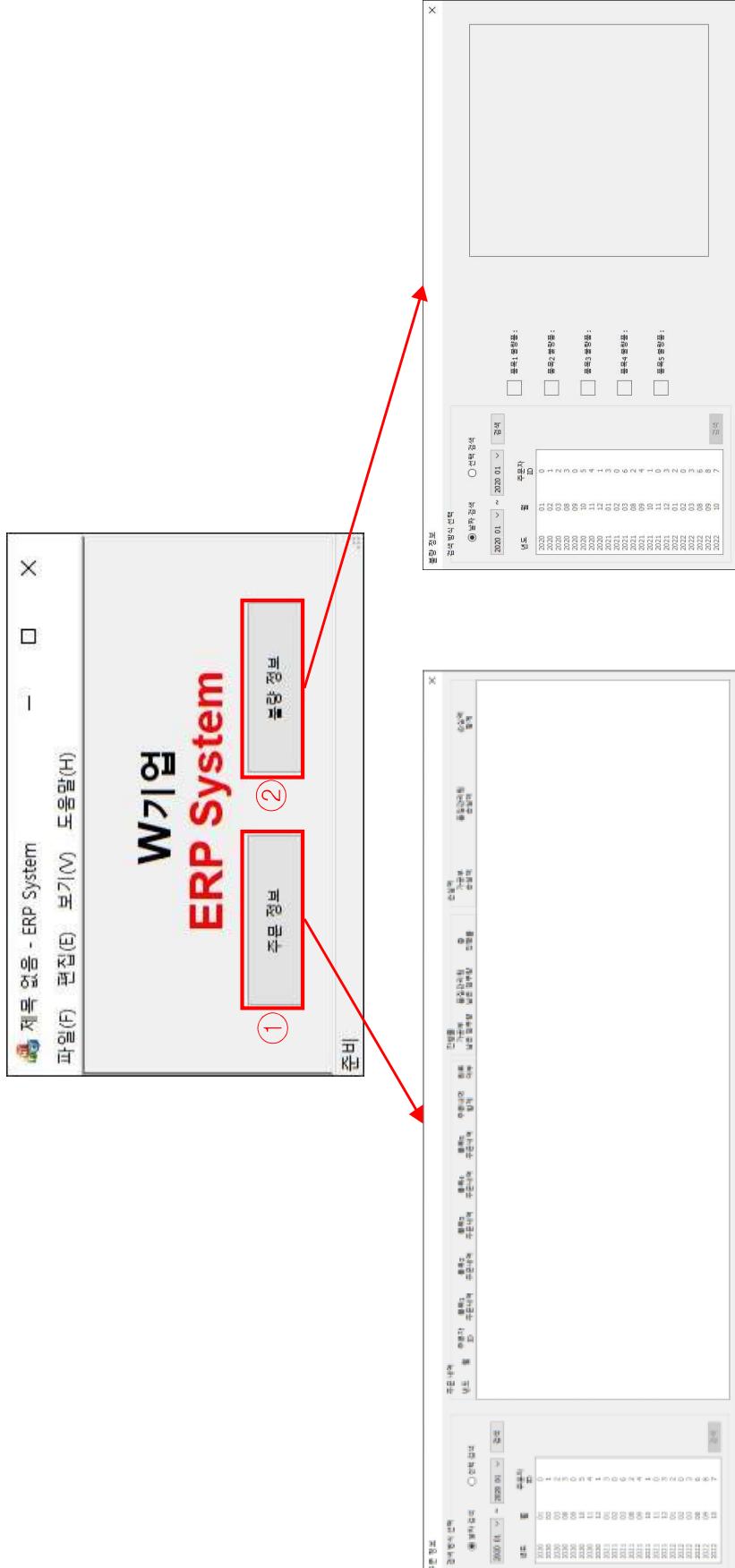


① : 각 품목 색상 (Picture Control)

② : 각 품목 불량품 개수 및 불량률 (Static Text)

③ : 불량률 원형 그래프 (Picture Control)

ERP System 기능 – 시작 Dialog



- ① : 주문 정보 Dialog 생성(주문 내역, 진행률, 손실액 등 출력)
 - ② : 불량 정보 Dialog 생성(품목별 불량품 개수 및 불량률 출력)

ERP System 기능 – 주문 정보 Dialog

주문 정보		검색 범위 선택									
<input type="radio"/> 날짜 검색	<input checked="" type="radio"/> 선택 검색	주문 내역	주문자 ID	품목1 주문내역	품목2 주문내역	품목3 주문내역	품목4 주문내역	품목5 주문내역	주문내역 합계	주문내역 합계	주문내역 합계
2020. 01. 01 ~	2020. 01. 01 ~			0	300	0	0	500	800	0	0
				400	0	100	1000	0	1500	0	0
				5	0	200	0	500	700	0	0
				12	1	0	100	0	200	800	0
				11	0	200	0	300	500	0	0
				02	0	300	0	200	200	0	0
				09	8	0	300	0	700	600	1600
				2022	10	7	500	0	800	1900	x
										736	795
										59%	58000
											₩ 540000

②		선택한 주문 내역									
주문 내역	주문자 ID	품목1 주문내역	품목2 주문내역	품목3 주문내역	품목4 주문내역	품목5 주문내역	주문내역 합계	주문내역 합계	주문내역 합계	주문내역 합계	주문내역 합계
2020 02 09 0	2020 09 0	0	300	0	0	500	800	0	0	0	0
2020 10 09 5	2020 10 5	400	0	100	1000	0	1500	0	0	700	0
2020 12 11 0	2020 12 1	0	200	0	500	0	200	500	0	800	0
2021 11 09 0	2021 11 0	200	0	300	0	300	0	0	1000	0	0
2022 02 09 0	2022 02 0	300	0	200	0	200	0	0	700	0	0
2022 09 08 0	2022 09 8	0	300	0	700	0	600	0	0	1600	x
2022 10 09 2	2022 10 7	500	0	600	0	600	0	0	1900	x	736
											795
											59%
											₩ 58000

① : Radio Button을 사용하여 날짜 검색 & 선택 검색으로 구분하여 검색

② : List Box(결과)에 검색된 기간에 대한 주문 내역, 진행률, 손해액 등 주문 정보를 출력

ERP System 기능 - 주문 정보 Dialog

① : 날짜 검색 선택 시 List Box(검색)와 선택한 Button B 활성화

② : 선택 검색 선택 시 Combo Box와 날짜 검색 Button 비활성화

③ : 날짜 검색 Button 클릭 시 |작 날짜가 끝 날짜 보다 클 경우 오류 메시지 출력

Etc. 각 Radio Button 클릭 시 List Box(검색)의 선택된 커서들 & List Box(출력) 초기화

ERP System 기능 – 주문 정보 Dialog

주문 내역									
번호	월	주문자 ID	품목1: 주문내역	품목2: 주문내역	품목3: 주문내역	품목4: 주문내역	품목5: 주문내역	주문내역 합계	주문내역
2021	03	6	300	0	200	300	0	800	0
2021	08	2	300	200	500	500	100	1600	0
2021	09	4	200	0	400	0	500	1100	0
2021	10	1	100	200	0	0	300	600	0
2021	11	0	200	0	300	500	0	1000	0
2021	12	3	100	300	400	0	0	800	0
2022	01	2	500	300	600	600	300	2300	0
2022	02	0	300	0	200	200	0	700	0
2022	03	3	200	500	200	0	0	900	0
2022	08	6	300	0	500	200	0	1000	x
2022	09	8	0	300	0	700	600	1600	x
2022	10	7	500	0	600	0	800	1900	x

진행률										
주문내역	기공부 남은 수량	여부	주문내역	기공부 남은 수량	여부	주문내역	기공부 남은 수량	여부	주문내역	
2021	80000	0	2021	185000	0	2021	621000	0	2021	701000
2021	148000	0	2021	148000	0	2021	1560000	0	2021	1745000
2021	50000	0	2021	50000	0	2021	880000	0	2021	1028000
2021	458000	0	2021	458000	0	2021	458000	0	2021	508000
2021	86000	0	2021	86000	0	2021	710000	0	2021	736000
2021	96000	0	2021	96000	0	2021	1174000	0	2021	1270000
2022	128000	0	2022	128000	0	2022	899000	0	2022	1027000
2022	24000	0	2022	24000	0	2022	464000	0	2022	4680000
2022	43000	0	2022	43000	0	2022	952000	0	2022	995000
2022	30000	0	2022	30000	0	2022	311000	0	2022	341000
2022	70000	0	2022	70000	0	2022	643000	0	2022	713000
2022	58000	0	2022	58000	0	2022	540000	0	2022	598000

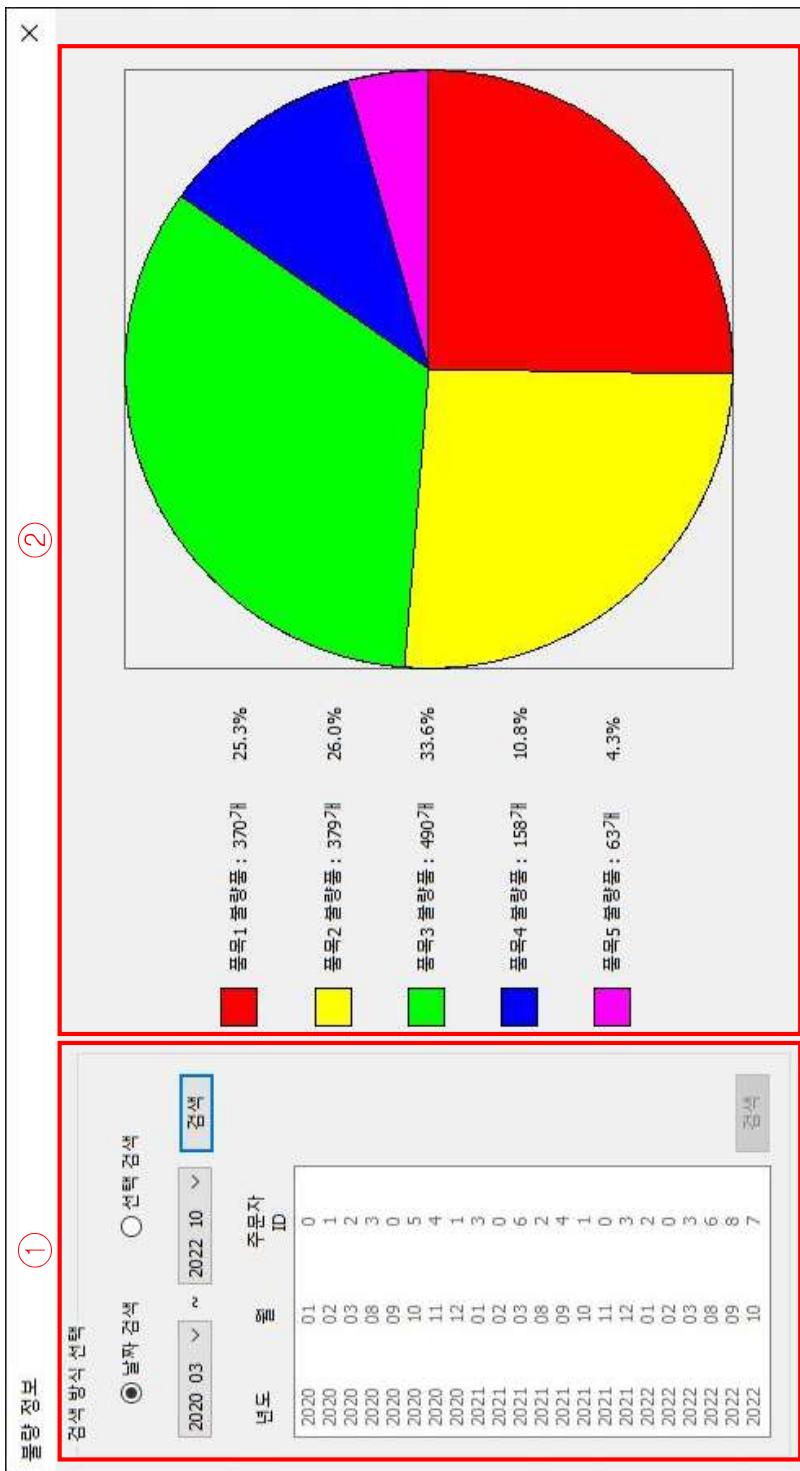
손실액										
주문내역	기공부 순상량	여부	주문내역	기공부 순상량	여부	주문내역	기공부 순상량	여부	주문내역	
2021	80000	0	2021	185000	0	2021	621000	0	2021	701000
2021	148000	0	2021	148000	0	2021	1560000	0	2021	1745000
2021	50000	0	2021	50000	0	2021	880000	0	2021	1028000
2021	458000	0	2021	458000	0	2021	458000	0	2021	508000
2021	86000	0	2021	86000	0	2021	710000	0	2021	736000
2021	96000	0	2021	96000	0	2021	1174000	0	2021	1270000
2022	128000	0	2022	128000	0	2022	899000	0	2022	1027000
2022	24000	0	2022	24000	0	2022	464000	0	2022	4680000
2022	43000	0	2022	43000	0	2022	952000	0	2022	995000
2022	30000	0	2022	30000	0	2022	311000	0	2022	341000
2022	70000	0	2022	70000	0	2022	643000	0	2022	713000
2022	58000	0	2022	58000	0	2022	540000	0	2022	598000

① : 주문 내역 - 년도, 월, 주문자 ID, 각 품목의 주문내역 및 주문내역 합, 완료 여부

② : 진행률 - 가공부와 품질관리팀의 남은 업무량 및 총 진행률

③ : 손실액 - 가공부와 품질관리팀에서 발견한 불량품의 손실액 및 총 손실액

ERP System 기능 - 물량 정보 Dialog



① : Radio Button을 사용하여 날짜 검색 & 선택 검색으로 구분하여 검색

② : 각 품목별 물량품 개수와 물량률을 Text와 원형 그래프로 출력

코드 구현

<파일 입력>

```
void CallData::GetData()
{
    ifstream fin;
    fin.open("IC-PBLdata.txt");
    int i = 0;
    int j = 0;
    while (!fin.eof()) {
        fin >> read_val[i][j];
        j++;
        if (j == 28) {
            i++;
            j = 0;
        }
    }
}
```

<Combo Box & List Box 테이터 입력>

```
void CallData::FillComboBox()
{
    for (int i = 0; i < 22; i++) {
        CString str;
        str.Format(_T("%d %02d"), read_val[i][0], read_val[i][1]);
        m_start.AddString(str);
        m_end.AddString(str);
    }
}

void CallData::FillSearchListBox()
{
    m_search_list.ResetContent();
    for (int i = 0; i < 22; i++) {
        CString str;
        str.Format(_T("%d %02d %d"), read_val[i][0], read_val[i][1], read_val[i][2]);
        m_search_list.AddString(str);
    }
}
```

코드 구현

<주문 정보 검색>

```
void CallData::OnBnClickedButton1()
{
    m_result_list.ResetContent();
    int start = m_start.GetCurSel();
    int end = m_end.GetCurSel();
    if (start > end) {
        MessageBox(_T("시작하는 날짜가 끝나는 날짜보다 빨라야합니다."), _T("오류"), MB_OK | MB_ICONERROR);
    }
    else {
        for (int i = start; i <= end; i++) {
            FillResultListBox(i);
        }
    }
}

void CallData::OnBnClickedButton2()
{
    m_result_list.ResetContent();
    int cnt = m_search_list.GetCount();
    for (int i = 0; i < cnt; i++) {
        if (m_search_list.GetSel(i)) {
            FillResultListBox(i);
        }
    }
}
```

한국

<주문 정보>

한국

<불량정보 검색>

```

void CEachData::OnBnClickedButton1()
{
    int start = m_start.GetCurSel();
    int end = m_end.GetCurSel();

    sum_err[0] = 0;
    sum_err[1] = 0;
    sum_err[2] = 0;
    sum_err[3] = 0;
    sum_err[4] = 0;

    if (start > end) {
        MessageBox(_T("시작하는 날짜가 끝나는 날짜보다 빨라야합니다."), _T("오류"), MB_OK | MB_ICONERROR);
    }
    else {
        for (int i = start; i <= end; i++) {
            SumError(i);
            FillResultchart();
        }
    }
}

void CEachData::OnBnClickedButton2()
{
    int cnt = m_search_list.GetCount();

    sum_err[0] = 0;
    sum_err[1] = 0;
    sum_err[2] = 0;
    sum_err[3] = 0;
    sum_err[4] = 0;

    for (int i = 0; i < cnt; i++) {
        if (m_search_list.GetSel(i)) {
            SumError(i);
            FillResultchart();
        }
    }
}

```

코드 구현

<불량 정보 출력>

```
void CEachData::SumError( int i )
{
    for ( int j = 0; j < 5; j++ )
    {
        sum_err[j] += read_val[i][18+j] + read_val[i][23+j];
    }

    str.Format(_T("%d"), sum_err[0]);
    m_error_item1.SetWindowText(str + "개");
    str.Format(_T("%d"), sum_err[1]);
    m_error_item2.SetWindowText(str + "개");
    str.Format(_T("%d"), sum_err[2]);
    m_error_item3.SetWindowText(str + "개");
    str.Format(_T("%d"), sum_err[3]);
    m_error_item4.SetWindowText(str + "개");
    str.Format(_T("%d"), sum_err[4]);
    m_error_item5.SetWindowText(str + "개");

    total = sum_err[0] + sum_err[1] + sum_err[2] + sum_err[3] + sum_err[4];

    for ( int i = 0; i < 5; i++ )
    {
        per[i] = (float)(sum_err[i]) / total * 100;
    }

    str.Format(_T("%1f"), per[0]);
    m_per1.SetWindowText(str + "%");
    str.Format(_T("%1f"), per[1]);
    m_per2.SetWindowText(str + "%");
    str.Format(_T("%1f"), per[2]);
    m_per3.SetWindowText(str + "%");
    str.Format(_T("%1f"), per[3]);
    m_per4.SetWindowText(str + "%");
    str.Format(_T("%1f"), per[4]);
    m_per5.SetWindowText(str + "%");
}
```

코드 구현

<원형 그라프 출력-1>

```
void CEachDate::F11ResultChart()
{
    x[0] = (int)width * cos((360 * per[0] / 100) * pi / 180)) + rectview.CenterPoint().x;
    y[0] = (int)height * sin((360 * per[0] / 100) * pi / 180)) + rectview.CenterPoint().y;

    x[1] = (int)width * cos((360 * (per[0] + per[1]) / 100) * pi / 180)) + rectview.CenterPoint().x;
    y[1] = (int)height * sin((360 * (per[0] + per[1]) / 100) * pi / 180)) + rectview.CenterPoint().y;

    x[2] = (int)width * cos((360 * (per[0] + per[1] + per[2]) / 100) * pi / 180)) + rectview.CenterPoint().x;
    y[2] = (int)height * sin((360 * (per[0] + per[1] + per[2]) / 100) * pi / 180)) + rectview.CenterPoint().y;

    x[3] = (int)width * cos((360 * (per[0] + per[1] + per[2] + per[3]) / 100) * pi / 180)) + rectview.CenterPoint().x;
    y[3] = (int)height * sin((360 * (per[0] + per[1] + per[2] + per[3]) / 100) * pi / 180)) + rectview.CenterPoint().y;

    CCClientDC dc(this);

    CBrush brush[5];
    brush[0].CreateSolidBrush(piColor[0]);
    brush[1].CreateSolidBrush(piColor[1]);
    brush[2].CreateSolidBrush(piColor[2]);
    brush[3].CreateSolidBrush(piColor[3]);
    brush[4].CreateSolidBrush(piColor[4]);

    dc.SelectObject(&brush[0]);
    dc.Rectangle(&rectcolor[1]);
    dc.SelectObject(&brush[1]);
    dc.Rectangle(&rectcolor[2]);
    dc.SelectObject(&brush[2]);
    dc.Rectangle(&rectcolor[3]);
    dc.SelectObject(&brush[3]);
    dc.Rectangle(&rectcolor[4]);
    dc.SelectObject(&brush[4]);
    dc.Rectangle(&rectcolor[5]);
}
```

코드 구현

<원형 그라프 출력-2>

```
if (sum_err[0])
{
    dc.SelectObject(&brush[0]);
}

if (sum_err[1])
{
    dc.SelectObject(&brush[1]);
    dc.Pie(rectview.TopLeft().y, rectview.TopLeft().x, rectview.BottomRight().y, rectview.BottomRight().x, rectview.BottomRight().y, init_y, init_x, init_y, x[0], y[0]);
}

if (sum_err[2])
{
    dc.SelectObject(&brush[2]);
    dc.Pie(rectview.TopLeft().y, rectview.BottomRight().x, rectview.BottomRight().y, init_y, init_x, init_y, x[1], y[1]);
}

if (sum_err[3])
{
    dc.SelectObject(&brush[3]);
    dc.Pie(rectview.TopLeft().y, rectview.BottomRight().x, rectview.BottomRight().y, init_y, init_x, init_y, x[2], y[2]);
}

if (sum_err[4])
{
    dc.SelectObject(&brush[4]);
    dc.Pie(rectview.TopLeft().y, rectview.BottomRight().x, rectview.BottomRight().y, init_y, init_x, init_y, x[3], y[3]);
}
```

동작 영상

The Engine of Korea

중간재 소통, 경쟁력 향상의 첨단 기술을 적용한
한양대 ERICA

