





## 당신의 소득, 세금으로 얼마나 내고 있을까요? ( 연말정산 도우미 )

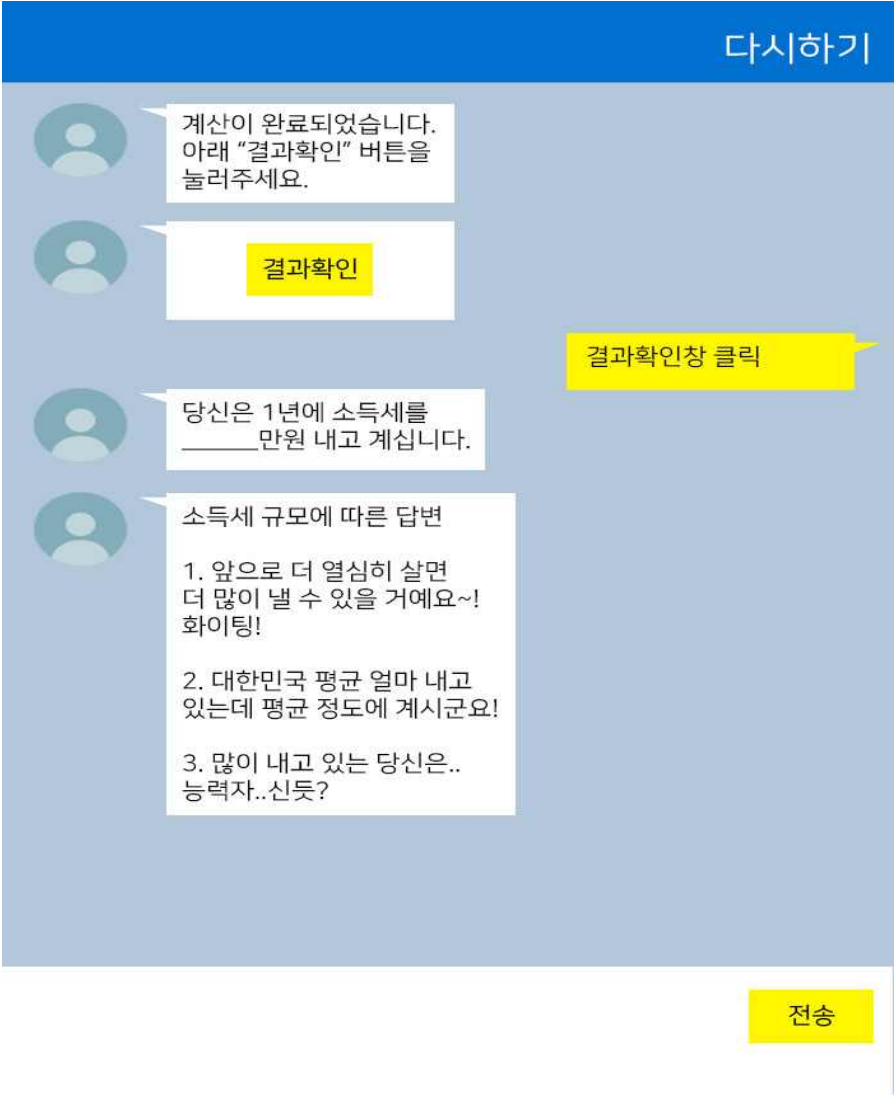
목적	근로자에게 1년 동안 소득세로 납부하는 금액이 대략 얼마인지 계산해주고, 원하는 사람에 한해서 일정한 정보를 입력하면, 연말정산시 얼마의 세금이 줄어드는지 확인시켜주는 것이 목적
----	--

	주요 흐름	세부 흐름
O	1. <a href="#">왓슨의 인사</a>	- 안내사항 제시 (대략적인 계산을 위한 것)
	2. <a href="#">대략적인 세액 계산</a>	
	2-1. <a href="#">계산을 원하는 연봉 입력 받기</a>	
	2-2. <a href="#">납부해야할 세금 금액 계산</a>	
	2-3. <a href="#">(결과확인) 버튼을 누르면 세액을 보여줌</a>	
X	3. <a href="#">구체적인 계산을 할 것인지 물어보기</a>	- 한다면 4번 / 안하면 7번으로 이동
	4. <a href="#">구체적인 세액을 위한 자료 입력 도입부 (자동 / 수동)</a>	
	<a href="#">4-1. 자료 있는지 여부 물어보기</a>	- 구체적인 세액 계산시, (국세청 기본자료 + 원천징수명세서) 가 필요하다는 점을 표시해줌. - 자료가 준비되었다면 4-1-1번로 이동 - 자료가 없다면 7번로 이동
	<a href="#">4-1-1. 사전 설문조사</a>	- 자료 입력 전 설문조사 진행 / 총 ____가지 진행 예정
	<a href="#">4-2. 해당 자료를 업로드 요청</a>	- 업로드할 자료가 없다면 4-5번로 이동
	<a href="#">4-3. 해당 자료 추출하여 계산식에 자동 입력하기</a>	- 자료 추출은 국세청 제공 JSP코드를 활용하여 XML 변환 / 자료 저장 추출
	<a href="#">4-4. 업로드 성공 / 실패 여부 표시해주기</a>	- 업로드 성공시 5번으로 이동 - 업로드 실패시 4-5번로 이동
	<a href="#">4-5. 수동입력 항목 입력 받기</a>	- 입력해야할 개수를 제시하고, 위의 “나중에 입력하기” 버튼 만들기 - 한다면 4-6로 이동 - 안한다면 7로 이동
	<a href="#">4-6. 수동입력 자료 검증하기</a>	
	<a href="#">5. 세액의 계산</a>	- 계산하기 버튼을 누르면 계산이 시작됨.
	<a href="#">6. 이용자가 (결과확인) 버튼을 누르면 세액을 표시해줌</a>	
O	<a href="#">7. 프로그램 종료 + 기타 참고자료 제시</a>	

번호	1. 왓슨의 인사	서비스 로직	
1	<div><div>다시하기</div><div><div><div>안녕하세요. 당신이 1년에 소득세로 세금을 얼마나 내는지 궁금하신가요?</div><div>1년 연봉에 대한 일반적인 세액 계산과 이용자에 맞는 연말정산 결과를 보여드립니다.</div><div>원하시는 항목의 번호를 입력해 주세요. 1. 일반적인 세액계산 2. 이용자 맞춤 세액계산</div></div><div>1번 / 1 / 위에 것 2번 / 2 / 아래 것 둘 다 둘 다 싫어</div><div>전송</div></div></div>	<div>[ 챗봇 ]</div> <div>안녕하세요. 당신이 1년에 소득세로 세금을 얼마나 내는지 궁금하신가요? 1년 연봉에 대한 일반적인 세액 계산과 이용자에 맞는 연말정산 결과를 보여드립니다. 원하시는 항목의 번호를 입력해 주세요.</div> <div>[ 선택 ]</div> <div>1. 일반적인 세액계산 2. 이용자 맞춤 세액계산</div>	
		<div>[ 유저 : 대답 ]</div> <div>1. 1번 / 1 / 위의 것 2. 2번 / 2 / 아래 것 3. 둘 다 4. 둘 다 싫어</div>	<div>-&gt; 2번으로 이동 -&gt; 4번으로 이동 -&gt; 2번으로 이동 -&gt; 7번으로 이동 (프로그램 종료 쪽)</div>
<div>[ 챗봇 : 유저 대답에 대한 반응]</div> <div>1. 일반적인 세액 계산을 선택하셨군요. 2. 이용자 맞춤 세액 계산을 선택하셨군요. 3. 두 개 다 선택하셨군요. 그럼 먼저 일반적인 세액계산을 먼저 진행하겠습니다. 4. 그럼 다음 기회에 함께 하시죠!</div>			
비고			

번호	2. 대략적인 세액 계산 - 2-1. 계산을 원하는 연봉 입력 받기	서비스 로직	
2 (2-1)	<div><div>다시하기</div><div><div><div><div></div><div>그럼 일반적으로 적용 받을 수 있는 항목을 바탕으로 당신의 1년치 연봉에 대한 소득세 계산을 해보겠습니다.</div></div><div><div></div><div>당신의 1년 연봉을 입력해주세요. (만원을 제외한 숫자만 입력해주세요.)</div></div><div><div>3415</div></div><div><div></div><div>3415만원에 대한 소득세를 알고싶으시군요.</div></div><div><div></div><div>잠시만 기다려주세요. 금방 계산결과를 알려드리겠습니다.</div></div></div><div>전송</div></div></div>	<div>[ 챗봇 ]</div> <div>그럼 일반적으로 적용받을 수 있는 항목을 바탕으로 당신의 1년치 연봉에 대한 소득세 계산을 해보겠습니다. 당신의 1년 연봉을 입력해주세요. (만원을 제외한 숫자만 입력해주세요.)</div>	
		<div>[ 유저 : 대답 ]</div> <div>3400 4500 3400만원</div>	<div>-&gt; -&gt; -&gt; (만원을 제외한 숫자만 입력 받음)</div>
		<div>[ 챗봇 : 유저 대답에 대한 반응]</div> <div>~~~~만원에 대한 소득세를 알고싶으시군요. 잠시만 기다려주세요. 금방 계산결과를 알려드리겠습니다.</div>	
비고			

번호	2. 대략적인 세액 계산 - 2-2 납부해야할 세금 금액 계산		서비스 로직 [ 계산 근거자료 ]													
2 (2-2)		근로소득공제	[ 계산식 ] X = 총급여액													
			<table><tr><th>조건 (총급여액이)</th><th>근로소득공제액</th></tr><tr><td>500만원 이하</td><td>X * 0.7</td></tr><tr><td>500만원 초과 1500만원 이하</td><td>350만원 + (X - 500만원) * 0.4</td></tr><tr><td>1500만원 초과 4500만원 이하</td><td>750만원 + (X - 1500만원) * 0.15</td></tr><tr><td>4500만원 초과 1억원 이하</td><td>1200만원 + (X - 4500만원) * 0.05</td></tr><tr><td>1억원 초과</td><td>1475만원 + (X - 1억원) * 0.02</td></tr></table>	조건 (총급여액이)	근로소득공제액	500만원 이하	X * 0.7	500만원 초과 1500만원 이하	350만원 + (X - 500만원) * 0.4	1500만원 초과 4500만원 이하	750만원 + (X - 1500만원) * 0.15	4500만원 초과 1억원 이하	1200만원 + (X - 4500만원) * 0.05	1억원 초과	1475만원 + (X - 1억원) * 0.02	
	조건 (총급여액이)	근로소득공제액														
	500만원 이하	X * 0.7														
	500만원 초과 1500만원 이하	350만원 + (X - 500만원) * 0.4														
	1500만원 초과 4500만원 이하	750만원 + (X - 1500만원) * 0.15														
	4500만원 초과 1억원 이하	1200만원 + (X - 4500만원) * 0.05														
1억원 초과	1475만원 + (X - 1억원) * 0.02															
[단위 : 만원]  총급여액 [입력금액] - 근로소득공제 = 근로소득금액	연금보험료 공제	[ 계산식 ]  총급여액 / 12 = 기준소득월액  기준소득월액 * 0.045 = 연금보험료 공제액														
		<table><tr><th>조건 ( 기준소득월액이 )</th><th>기준소득월액</th></tr><tr><td>기준소득월액 &lt; 29</td><td>29</td></tr><tr><td>29 =&lt; 기준소득월액 =&lt; 449</td><td>기준소득월액</td></tr><tr><td>449 &lt; 기준소득월액</td><td>449</td></tr></table>	조건 ( 기준소득월액이 )	기준소득월액	기준소득월액 < 29	29	29 =< 기준소득월액 =< 449	기준소득월액	449 < 기준소득월액	449						
		조건 ( 기준소득월액이 )	기준소득월액													
기준소득월액 < 29	29															
29 =< 기준소득월액 =< 449	기준소득월액															
449 < 기준소득월액	449															
- 연금보험료 공제 = 종합소득 과세표준  x 기본세율 = 산출세액	기본세율	<table><tr><th>조건 ( 종합소득 과세표준이 )</th><th>산출세액</th></tr><tr><td>1200만원 이하</td><td>( 종합소득 과세표준 * 0.06 )</td></tr><tr><td>1200만원 초과 4600만원 이하</td><td>( " * 0.15 ) - 108</td></tr><tr><td>4600만원 초과 8800만원 이하</td><td>( " * 0.24 ) - 552</td></tr><tr><td>8800만원 초과 15000만원 이하</td><td>( " * 0.35 ) - 1490</td></tr><tr><td>15000만원 초과 50000만원 이하</td><td>( " * 0.38 ) - 1940</td></tr><tr><td>50000만원 초과</td><td>( " * 0.40 ) - 2940</td></tr></table>	조건 ( 종합소득 과세표준이 )	산출세액	1200만원 이하	( 종합소득 과세표준 * 0.06 )	1200만원 초과 4600만원 이하	( " * 0.15 ) - 108	4600만원 초과 8800만원 이하	( " * 0.24 ) - 552	8800만원 초과 15000만원 이하	( " * 0.35 ) - 1490	15000만원 초과 50000만원 이하	( " * 0.38 ) - 1940	50000만원 초과	( " * 0.40 ) - 2940
		조건 ( 종합소득 과세표준이 )	산출세액													
		1200만원 이하	( 종합소득 과세표준 * 0.06 )													
		1200만원 초과 4600만원 이하	( " * 0.15 ) - 108													
		4600만원 초과 8800만원 이하	( " * 0.24 ) - 552													
		8800만원 초과 15000만원 이하	( " * 0.35 ) - 1490													
15000만원 초과 50000만원 이하	( " * 0.38 ) - 1940															
50000만원 초과	( " * 0.40 ) - 2940															
- 근로소득 세액공제 - 표준세액공제 = 결정세액	근로소득 세액공제	조건1 :														
		<table><tr><th>조건 ( 산출세액이 )</th><th>근로소득 세액공제액</th></tr><tr><td>130만원 이하</td><td>( 산출세액 * 0.55 )</td></tr><tr><td>130만원 초과</td><td>( 산출세액 - 130만원 ) * 0.30 + 71.5만원</td></tr></table>	조건 ( 산출세액이 )	근로소득 세액공제액	130만원 이하	( 산출세액 * 0.55 )	130만원 초과	( 산출세액 - 130만원 ) * 0.30 + 71.5만원								
		조건 ( 산출세액이 )	근로소득 세액공제액													
		130만원 이하	( 산출세액 * 0.55 )													
		130만원 초과	( 산출세액 - 130만원 ) * 0.30 + 71.5만원													
조건2 :																
<table><tr><th>조건 ( 총급여액이 )</th><th>한도액</th></tr><tr><td>총급여액 =&lt; 3300</td><td>74만원</td></tr><tr><td>3300 &lt; 총급여액 =&lt; 7000만원</td><td>max ( 74만원 - (총급여액 - 3300만원)*0.008, 66만원)</td></tr><tr><td>7000 &lt; 총급여액</td><td>max ( 66만원 - (총급여액 - 7000만원)* 0.5, 50만원)</td></tr></table>	조건 ( 총급여액이 )	한도액	총급여액 =< 3300	74만원	3300 < 총급여액 =< 7000만원	max ( 74만원 - (총급여액 - 3300만원)*0.008, 66만원)	7000 < 총급여액	max ( 66만원 - (총급여액 - 7000만원)* 0.5, 50만원)								
조건 ( 총급여액이 )	한도액															
총급여액 =< 3300	74만원															
3300 < 총급여액 =< 7000만원	max ( 74만원 - (총급여액 - 3300만원)*0.008, 66만원)															
7000 < 총급여액	max ( 66만원 - (총급여액 - 7000만원)* 0.5, 50만원)															
표준세액공제	13만원 ( 특별소득공제, 특별세액공제, 월세액세액공제를 신청하지 않음)															

번호	2. 대략적인 세액 계산 - 2-3 (결과확인) 버튼을 누르면 세액을 보여줌	서비스 로직
2 (2-3)		<p>[ 챗봇 ] :</p> <p>계산이 완료되었습니다. 아래 “결과확인” 버튼을 눌러주세요. “결과확인”</p>
		<p>[ 유저 : 대답 ] : 결과 확인창 클릭</p>
		<p>[ 챗봇 ]</p> <p>당신은 1년에 소득세를 ____만원 내고 계십니다. [ 소득세 규모에 따른 답변 ]</p> <ol style="list-style-type: none"> <li>1. 소 : 앞으로 더 열심히 살면 더 많이 낼 수 있을 거예요~~! 화이팅!</li> <li>2. 중 : 대한민국 평균 정도 내고 계시군요!</li> <li>3. 대 : 많이 내고 있는 당신은.. 능력자.. 신 듯?</li> </ol>
		비고

번호	3. 구체적인 계산을 할 것인지 물어보기	서비스 로직	
3	<div><div>다시하기</div><div><div><div><div></div><div>결과를 확인해보셨는데, 구체적으로 계산해 보시겠 습니까?</div></div><div><div>1. 예 / 네 / 그래요 2. 아니오 / 노 / ㄴㄴ</div></div></div><div><div><div><div></div><div>1. 그럼 구체적으로 계산해 보러 가시죠. 2. 그럼 다음 기회에 한번 해봐요~!</div></div><div><div>전송</div></div></div></div></div></div>	<div>[ 챗봇 ] : 결과를 확인해보셨는데, 구체적으로 계산해 보시겠습니까?</div>	
		<div>[ 유저 : 대답 ] 1. 예 / 네 / 그래요 2. 아니오 / 노 / ㄴㄴ</div>	<div>-&gt; 4번으로 이동 -&gt; 7번으로 이동 (프로그램 종료 쪽)</div>
		<div>[ 챗봇 ] : 1. 그럼 구체적으로 확인하러 가시죠. 2. 그럼 다음 기회에 한번 해봐요~!</div>	
비고			

번호	7. 프로그램 종료 + 기타 참고자료 제시	서비스 로직
7		<p>[ 챗봇 ]</p> <p>결과를 확인해보시니 어떠신가요? 다시 계산해보고 싶으시다면, 아래 위에 “처음부터 다시하기” 버튼을 눌러주시고, 연말정산 관련한 체크리스트 등 자료를 보내드리니, 궁금하시면 참고해주세요.</p> <p>세금은 준비하는 자만이 아낄 수 있습니다. 궁금하신 사항은 가까운 세무사를 찾아주세요! 감사합니다.</p>
		비고

```
1  package com.test.income;
2
3  import org.springframework.stereotype.Controller;
4  import org.springframework.web.bind.annotation.RequestMapping;
5
6  @Controller
7  @RequestMapping("chatbot")
8  public class ChatbotController {
9
10     @RequestMapping("/")
11     public String init() {
12         return "chatbot";
13     }
14 }
15 /*
16
17     @RequestMapping()
18     // 왓슨과의 대화한 것을 가로챌 메서드
19
20     @RequestMapping()
21     // 계산 한 값을 가져와서 뷰에 뿌려줄 전송할 메서드
22 */
23 }
```



```
1     package com.test.income;
2
3     import java.text.DateFormat;
4     import java.util.Date;
5     import java.util.Locale;
6
7     import org.slf4j.Logger;
8     import org.slf4j.LoggerFactory;
9     import org.springframework.stereotype.Controller;
10    import org.springframework.ui.Model;
11    import org.springframework.web.bind.annotation.RequestMapping;
12    import org.springframework.web.bind.annotation.RequestMethod;
13
14    /**
15     * Handles requests for the application home page.
16     */
17    @Controller
18    public class HomeController {
19
20        private static final Logger logger =
21            LoggerFactory.getLogger(HomeController.class);
22
23        /**
24         * Simply selects the home view to render by returning its name.
25         */
26        @RequestMapping(value = "/", method = RequestMethod.GET)
27        public String home(Locale locale, Model model) {
28            logger.info("Welcome home! The client locale is {}. ", locale);
29
30            Date date = new Date();
31            DateFormat dateFormat = DateFormat.getDateInstance(DateFormat.LONG,
32                DateFormat.LONG, locale);
33
34            String formattedDate = dateFormat.format(date);
35
36            model.addAttribute("serverTime", formattedDate );
37
38            return "home";
39        }
40    }
```

```

1  package com.test.income;
2
3  import javax.servlet.http.HttpSession;
4
5  import org.slf4j.Logger;
6  import org.slf4j.LoggerFactory;
7  import org.springframework.web.bind.annotation.RequestMapping;
8  import org.springframework.web.bind.annotation.RestController;
9
10 import com.ibm.watson.developer_cloud.conversation.v1.Conversation;
11 import com.ibm.watson.developer_cloud.conversation.v1.model.Context;
12 import com.ibm.watson.developer_cloud.conversation.v1.model.InputData;
13 import com.ibm.watson.developer_cloud.conversation.v1.model.MessageOptions;
14 import com.ibm.watson.developer_cloud.conversation.v1.model.MessageResponse;
15 import com.tax.income.calculator.TaxCalculator1;
16
17 @RestController
18 @RequestMapping("chatbot")
19 public class WatsonSay {    //Logger slf4j interface
20     private static final Logger logger = LoggerFactory.getLogger(WatsonSay.class);
21
22     @RequestMapping("watsonsays")
23     public MessageResponse watsonsays(String isay, HttpSession session) {
24         logger.info("user input : " + isay);
25
26         int result = -999;
27         int tax = -999;
28         int abcd;
29
30         // System.out.println(isay);
31         if(isay.equals("")) {
32             session isNew();
33
34             isay="안녕하세요.";
35         }
36
37         Conversation service = new Conversation(Conversation.VERSION_DATE_2017_05_26);
38         service.setUsernameAndPassword("0fbd5c7c-2bfb-4e49-95b3-c8e110ba40bb",
39             "zVNs4j4d30mG");
40
41         System.out.println();
42
43         InputData input = new InputData.Builder(isay).build();
44         MessageOptions options=null;
45
46         if(!isay.equals("")) {
47             options =
48                 new MessageOptions.Builder("3d479c3a-cf43-4aa0-a256-9a4c690a7c27")
49                     .input(input)
50                     .context((Context)session.getAttribute("context"))
51                     .build();
52         } else {
53             options =
54                 new MessageOptions.Builder("3d479c3a-cf43-4aa0-a256-9a4c690a7c27") //
55                     .input(input)
56                     .context((Context)session.getAttribute("context"))
57                     .build();
58
59             MessageResponse response = service.message(options).execute();
60             // 응답에 대한 모든 답을 갖고 있음 -> 이 것이 jsp로 넘어감. 내 결과값을 response에 같이 붙여서
61             보낸다....
62
63             Context context = response.getContext();
64             logger.info("salary1 ::::: "+context.get("salary1"));
65
66             session.setAttribute("context", context);
67             if (!context.get("salary1").toString().equals("-999.0")) {
68                 int k = (int)Double.parseDouble(context.get("salary1").toString());
69                 tax = TaxCalculator1.main(k);
70                 response.put("tax", tax);

```

```
71         session.setAttribute("tax", tax);
72     }
73
74     String watsonsay = response.getOutput().getText().get(0);
75     logger.info("context ::::"+response.getContext().toString());
76
77
78     // System.out.println("result : "+ result);
79     // System.out.println("tax : "+ response.get("tax"));
80
81         // jsp에서 접근하는 값 //
82     System.out.println("watconsay : " + watsonsay); // watson이 말하는 것을 보여줌.
83     // logger.info(response.toString()); //로그파일
84     return response;
85 }
86 }
87
```

```

1  package com.tax.income.calculator;
2
3  public class TaxVO {
4      private static TaxVO tvo = null;
5      double salary1 = 0;      // 총급여액 일반
6      double inde1 = 0;        // 근로소득공제
7      double inde2 = 0;        // 연금보험료 공제
8      double multirate = 0;    // 기본세율
9      double taxcred1 = 0;     // 근로세액공제
10     double taxcred2 = 0;     // 표준세액공제
11
12     double afterinde1 = 0;    // 근로소득금액
13     double afterinde2 = 0;    // 종합소득과세표준
14     double aftermultirate = 0; // 산출세액
15     double finishvalue = 0;   // 결정세액 보여줄 값
16     // 소득공제 Incomededuction // 세액공제 Tax credit
17
18
19
20
21
22
23
24
25     public static TaxVO getInstance() {
26         if(tvo==null) {
27             tvo = new TaxVO();
28         }
29         return tvo;
30     }
31
32 }
33

```

```

1 package com.tax.income.calculator;
2
3 import java.util.Scanner;
4 import com.tax.income.calculator.Incomededuction1;
5 import com.tax.income.calculator.Incomededuction2;
6 import com.tax.income.calculator.Multirate;
7 import com.tax.income.calculator.TaxCred1;
8 import com.tax.income.calculator.TaxVO;
9
10 public class TaxCalculator1 {
11     public static int main(int k) {
12         // int 형 반환 / 인자를 받아들이м.
13         TaxVO tvo = TaxVO.getInstance();
14
15         System.out.println("1년치 급여를 입력해주세요.");
16         tvo.salary1 = k;
17
18         Incomededuction1 inde1 = new Incomededuction1();
19         double inde1result = inde1.inde1(tvo.salary1);
20         double afterinde1result = tvo.salary1 - inde1result;
21         // System.out.println(tvo.salary1);
22         // System.out.println(inde1result);
23         // System.out.println(afterinde1result);
24         // System.out.println();
25         // System.out.println(afterinde1result + " = " + tvo.salary1 + "-" +
inde1result);
26
27         System.out.println("      총급여액          : " + tvo.salary1);
28         System.out.println(" - 근로소득공제      : " + inde1result);
29         System.out.println(" = 근로소득금액          : " + afterinde1result);
30
31         Incomededuction2 inde2 = new Incomededuction2();
32         Incomededuction21 inde21 = new Incomededuction21();
33         double inde2result = (inde2.inde2(tvo.salary1)*0.045*12);
34         System.out.println(" 기준월소득액:" + inde2.inde2(tvo.salary1));
35         double inde21result = inde21.inde21(tvo.salary1*12);
36         double afterinde2result = afterinde1result - 150 - (inde2result +
inde21result);
37
38         if (afterinde2result < 0)
39             afterinde2result = 0;
40
41         System.out.println(" - 연금보험료공제      : " + 150);
42         System.out.println(" - 연금보험료공제      : " + inde2result);
43         System.out.println(" - 특별소득공제      : " + inde21result);
44         System.out.println(" = 종합소득 과세표준      : " + afterinde2result);
45
46         Multirate mulrate = new Multirate();
47         double multiresult = mulrate.mulrate(afterinde2result);
48
49         System.out.println(" = 산출세액          : " + multiresult);
50
51         TaxCred1 taxcred1 = new TaxCred1();
52         double taxcred1result = taxcred1.taxcred1(multiresult);
53
54         int finishvalue = (int) (multiresult - taxcred1result);
55         if (finishvalue < 0)
56             finishvalue = 0;
57
58         System.out.println("- 근로소득세액공제 : " + taxcred1result);
59         // System.out.println("- 표준세액공제      : " + 13);      - 특별소득공제 (고용, 건강보험으로
특별세액공제 적용 xx)
60         System.out.println(" = 결정세액          : " + finishvalue);
61         System.out.println();
62         System.out.println(" 당신이 1년에 납부하는 소득세는 " + finishvalue + "만원입니다.");
63
64         return finishvalue;
65     }
66 }
67

```

```

1 package com.tax.income.calculator;
2
3 public class Incomededuction1 {
4
5     public double indel(double salary1) {
6
7
8         if (salary1 <= 500)
9             return salary1 * 0.7;
10
11         else if (salary1 <= 1500 & salary1 > 500)
12             return (350 + (salary1 - 500) * 0.4);
13
14         else if (salary1 <= 4500 & salary1 > 1500)
15             return (750 + (salary1 - 1500) * 0.15);
16
17         else if (salary1 <= 10000 & salary1 > 4500)
18             return (1200 + (salary1 - 4500) * 0.05);
19
20         else if (salary1 > 10000)
21             return (1475 + (salary1 - 10000) * 0.02);
22
23         else
24             System.out.println("??? ?? 값을 ??? ??? ??? ??? ??? ???");
25         return salary1;
26     }
27
28
29 }

```

```
1 package com.tax.income.calculator;
2
3 public class Incomededuction2 {
4
5     public double inde2(double salary1) {
6
7
8         if (salary1/12 <29)
9             return 29;
10
11         else if (salary1/12 <= 449 & salary1/12 > 29)
12             return salary1/12;
13
14         else
15             return 449;
16     }
17
18
19 }
```

```
1 package com.tax.income.calculator;
2
3 public class Incomededuction21 {
4
5     public double inde21(double salary1) {
6
7
8         return (((salary1)/12 * (0.0065)) + ((salary1)/12*(0.0312)*(1.0738)));
9     }
10
11
12 }
```



```
1  package com.tax.income.calculator;
2
3  public class Multirate {
4
5      public double mulrate(double afterinde2result) {
6
7          if (afterinde2result <= 1200)
8              return (afterinde2result * 0.06);
9
10         else if (afterinde2result <= 4600 & afterinde2result > 1200)
11             return (afterinde2result * 0.15) - 108;
12
13         else if (afterinde2result <= 8800 & afterinde2result > 4600)
14             return (afterinde2result * 0.24) - 522;
15
16         else if (afterinde2result <= 15000 & afterinde2result > 8800)
17             return (afterinde2result * 0.35) - 1490;
18
19         else if (afterinde2result <= 50000 & afterinde2result > 15000)
20             return (afterinde2result * 0.38) - 1940;
21
22         else
23             return (afterinde2result * 0.40) - 2940;
24
25     }
26
27 }
28
```

```

1  package com.tax.income.calculator;
2
3  public class TaxCred1 {
4
5      public double taxcred1(double multiresult) {
6          TaxVO tvo = TaxVO.getInstance();
7          double a1 = 0;
8          double b1 = 0;
9          double c1 = 0;
10         double d1 = 0;
11
12
13         if (multiresult <= 130)
14             return multiresult * 0.55;
15
16         else
17             if( tvo.salary1 <= 3300 ) {
18                 a1 = 71.5 + (multiresult - 130) * 0.3;
19                 b1 = 74;
20                 System.out.println("        ** 근로소득공제 계산 : " + a1 );
21                 System.out.println("        ** 근로소득공제 한도액 : " + b1 );
22                 return (a1 > b1 ? b1 : a1);
23             }
24             else if (tvo.salary1 <= 7000 & tvo.salary1 > 3300 ) {
25                 a1 = 71.5 + (multiresult - 130) * 0.3;
26                 c1 = 74 - (tvo.salary1 - 3300) * 8/1000;
27                 d1 = 66;
28                 b1 = (c1 > d1 ? c1 : d1);
29                 System.out.println("        ** 근로소득공제 계산 : " + a1 );
30                 System.out.println("        ** 근로소득공제 한도액 : " + b1 );
31                 return (a1 > b1 ? b1 : a1);
32             }
33         }
34         else
35             a1 = 71.5 + (multiresult - 130) * 0.3;
36             c1 = 66 - (tvo.salary1 - 7000) * 1 / 2;
37             d1 = 50;
38             b1 = (c1 > d1 ? c1 : d1);
39             System.out.println("        ** 근로소득공제 계산 : " + a1 );
40             System.out.println("        ** 근로소득공제 한도액 : " + b1 );
41             return (a1 > b1 ? b1 : a1);
42     }
43 }

```

```

1  <%@ page language="java" contentType="text/html; charset=UTF-8"
2      pageEncoding="UTF-8"%>
3  <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
4  <!DOCTYPE html>
5  <html>
6  <head>
7  <meta charset="UTF-8">
8  <!-- bootstrap start -->
9  <meta http-equiv="X-UA-Compatible" content="IE=edge">
10 <meta name="viewport" content="width=device-width, initial-scale=1">
11 <title> page title </title>
12 <link href="<c:url value="/resources/css/bootstrap.min.css" />" rel="stylesheet">
13 <script src="<c:url value="/resources/js/jquery-3.3.1.js" />"></script>
14 <script src="<c:url value="/resources/js/bootstrap.min.js" />"></script>
15 <!-- bootstrap end -->
16 <link href="<c:url value="/resources/css/bubbles.css" />" rel="stylesheet">
17
18 <script type="text/javascript">
19 $(document).ready(function() {
20     $('#btn_restart').click(function() {
21         let re = confirm('모든 내용이 초기화됩니다. \n 정말 재시작 할까요?');
22         if(re) {
23             $('#said').html('');
24             ajax_process('');
25         }
26     });
27
28     $('#txt_isay').keypress(function(e) {
29         if(e.which==13) {
30             conversation();
31         }
32     });
33
34     $('#btn_isay').click(function(e) {
35         conversation();
36     });
37     ajax_process('');
38 });
39
40 function conversation() {
41     let _isay = $.trim($('#txt_isay').val());
42     if(_isay=='') {
43         alert('내용을 입력하세요');
44         return;
45     }
46     let isay = '<div class="row" col="5">';
47
48     isay += '<p class="col-lg-3 col-sm-3">'
49     //triangle-border right
50     isay += '<p class="triangle-isosceles right col-lg-5 col-sm-5">' +
51     '<font size="3">' + _isay + '</font>'
52     + '</p>';
53     isay += ''
55     width="100"/>';
56     isay += '</div>';
57     $('#said').append(isay);
58     $('#txt_isay').val('');
59     ajax_process(_isay);
60 }
61
62 function ajax_process(_isay) {
63     $.ajax({
64         type: 'POST',
65         url: 'watsonsay',
66         data: { 'isay': _isay },
67         async: true, //비동기: 거짓, 즉 저거는 동기다.
68         success: function(data) {
69             let watsonsay = '<div class="row">';
70             //watsonsay += '<p class="col-lg-1">';
71             watsonsay += '' width="100"/>';

```

```

72         watsonsay +=
73             //'<div class="col-lg-6">' +
74             '<p class="triangle-isosceles left col-lg-5 col-sm-5">' +
75             '<font size="3">' + '<br>';
76
77         console.log('context : '+data.tax);
78         console.log('abcd.context : '+data.context.abcd);
79         console.log('abcd : '+data.abcd);
80
81         if(data.context.abcd == 1) {
82             // if(data.tax != undefined){
83                 watsonsay += '<br>' + '납부하는 세금 : 약 '+data.tax+' 만원 ';
84             };
85             watsonsay += data.output.text;
86             watsonsay += '</font>' + '</p>';
87             //'</div>';
88             watsonsay += '</div>';
89             watsonsay = watsonsay.replace(/_/g, "<br/>");
90             watsonsay = watsonsay.replace(/,/g, "");
91             $('#said').append(watsonsay);
92             $('html, body').animate({scrollerTop: $(document).height()}, 500);
93
94             /*             '<font size="3">' + '<br>' + data.output.text;
95
96             console.log('context : '+data.tax);
97             console.log('abcd.context : '+data.context.abcd);
98             console.log('abcd : '+data.abcd);
99
100             if(data.context.abcd == 1) {
101                 // if(data.tax != undefined){
102                     watsonsay += '<br>' + '결과는 '+data.tax+' 만원으로 확인되었습니다.';
103                 };
104             */
105
106             if(data.context.req == 'list'){
107                 ajax_process_getList(data);
108             } else if(data.context.req == 'info'){
109                 ajax_process_getInfo(data.context.Name);
110             }
111             //html body에서 현재 문서의 스크롤의 top는 문서의 높이만큼이다
112             //내려가는 시간은 0.5초이다.
113         }
114     }); //ajax 괄호 끝
115 }
116 </script>
117 </head>
118 <body>
119 <div id="said"></div>
120 <input type="text" id="txt_isay" class="triangle-border center col-lg-6"
121 autofocus="autofocus"/>
122 <button type="button" id="btn_isay" class="btn btn-default col-lg-2">입력</button>
123 <button type="button" id="btn_restart" class="btn btn-default">대화 재시작</button>
124 </body>
</html>

```

```

1  <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
2  <%@ page session="false" pageEncoding="utf-8"%>
3  <html>
4  <head>
5  <title>근로소득전용 세액계산</title>
6  </head>
7  <style>
8  p {
9      font-size: 25px;
10 }
11
12 p.thick {
13     font-weight: bold;
14 }
15 </style>
16
17 <body>
18     <div class="entry" align="center">
19         <h2>
20             <br>1년 열심히 일해서 받은 당신의 연봉<br>
21         </h2>
22         <h2>당신은</h2>
23         <h2 style="color: red;">얼마의 소득세를</h2>
24         <h2>내고 계신지 알고 있습니까?<br>
25         </h2>
26         
28         <ol>
29             <li>
30                 <p class="thick" style="color: red;" align="center">
31                     <a href="chatbot/">>>>>>>>>>>>>>>>>>>>>>>>>>>></a>
32                 </p>
33             </li>
34         </ol>
35         <p class="thick">The time on the server is ${serverTime}.</P>
36     </div>
37 </body>
38 </html>
39

```

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <project xmlns="http://maven.apache.org/POM/4.0.0"
3  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
5  http://maven.apache.org/maven-v4_0_0.xsd">
6  <modelVersion>4.0.0</modelVersion>
7  <groupId>com.test</groupId>
8  <artifactId>test2</artifactId>
9  <name>test</name>
10 <packaging>war</packaging>
11 <version>1.0.0-BUILD-SNAPSHOT</version>
12 <properties>
13   <java-version>1.6</java-version>
14   <org.springframework-version>3.1.1.RELEASE</org.springframework-version>
15   <org.aspectj-version>1.6.10</org.aspectj-version>
16   <org.slf4j-version>1.6.6</org.slf4j-version>
17 </properties>
18 <dependencies>
19   <!-- Spring -->
20   <dependency>
21     <groupId>org.springframework</groupId>
22     <artifactId>spring-context</artifactId>
23     <version>${org.springframework-version}</version>
24     <exclusions>
25       <!-- Exclude Commons Logging in favor of SLF4j -->
26       <exclusion>
27         <groupId>commons-logging</groupId>
28         <artifactId>commons-logging</artifactId>
29       </exclusion>
30     </exclusions>
31   </dependency>
32   <dependency>
33     <groupId>org.springframework</groupId>
34     <artifactId>spring-webmvc</artifactId>
35     <version>${org.springframework-version}</version>
36   </dependency>
37   <!-- AspectJ -->
38   <dependency>
39     <groupId>org.aspectj</groupId>
40     <artifactId>aspectjrt</artifactId>
41     <version>${org.aspectj-version}</version>
42   </dependency>
43   <!-- Logging -->
44   <dependency>
45     <groupId>org.slf4j</groupId>
46     <artifactId>slf4j-api</artifactId>
47     <version>${org.slf4j-version}</version>
48   </dependency>
49   <dependency>
50     <groupId>org.slf4j</groupId>
51     <artifactId>jcl-over-slf4j</artifactId>
52     <version>${org.slf4j-version}</version>
53     <scope>runtime</scope>
54   </dependency>
55   <dependency>
56     <groupId>org.slf4j</groupId>
57     <artifactId>slf4j-log4j12</artifactId>
58     <version>${org.slf4j-version}</version>
59     <scope>runtime</scope>
60   </dependency>
61   <dependency>
62     <groupId>log4j</groupId>
63     <artifactId>log4j</artifactId>
64     <version>1.2.15</version>
65     <exclusions>
66       <exclusion>
67         <groupId>javax.mail</groupId>
68         <artifactId>mail</artifactId>
69       </exclusion>
70     </exclusions>
71     <groupId>javax.jms</groupId>
```

```

72         <artifactId>jms</artifactId>
73     </exclusion>
74     <exclusion>
75         <groupId>com.sun.jdmk</groupId>
76         <artifactId>jmxtools</artifactId>
77     </exclusion>
78     <exclusion>
79         <groupId>com.sun.jmx</groupId>
80         <artifactId>jmxri</artifactId>
81     </exclusion>
82 </exclusions>
83     <scope>runtime</scope>
84 </dependency>
85
86 <!-- https://mvnrepository.com/artifact/log4j/log4j -->
87 <dependency>
88     <groupId>log4j</groupId>
89     <artifactId>log4j</artifactId>
90     <version>1.2.17</version>
91 </dependency>
92
93 <!-- @Inject -->
94 <dependency>
95     <groupId>javax.inject</groupId>
96     <artifactId>javax.inject</artifactId>
97     <version>1</version>
98 </dependency>
99
100 <!-- Servlet -->
101 <dependency>
102     <groupId>javax.servlet</groupId>
103     <artifactId>servlet-api</artifactId>
104     <version>2.5</version>
105     <scope>provided</scope>
106 </dependency>
107
108 <dependency>
109     <groupId>javax.servlet.jsp</groupId>
110     <artifactId>jsp-api</artifactId>
111     <version>2.1</version>
112     <scope>provided</scope>
113 </dependency>
114 <dependency>
115     <groupId>javax.servlet</groupId>
116     <artifactId>jstl</artifactId>
117     <version>1.2</version>
118 </dependency>
119
120 <!-- Test -->
121 <dependency>
122     <groupId>junit</groupId>
123     <artifactId>junit</artifactId>
124     <version>4.7</version>
125     <scope>test</scope>
126 </dependency>
127
128 <!-- watson java sdk -->
129 <dependency>
130     <groupId>com.ibm.watson.developer_cloud</groupId>
131     <artifactId>java-sdk</artifactId>
132     <version>4.2.1</version>
133 </dependency>
134
135 <!-- https://mvnrepository.com/artifact/org.springframework/spring-context -->
136 <dependency>
137     <groupId>org.springframework</groupId>
138     <artifactId>spring-context</artifactId>
139     <version>5.0.4.RELEASE</version>
140 </dependency>
141
142 <!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->
143 <dependency>
144     <groupId>org.springframework</groupId>

```

```

145         <artifactId>spring-webmvc</artifactId>
146         <version>5.0.4.RELEASE</version>
147     </dependency>
148
149
150 </dependencies>
151 <build>
152     <plugins>
153         <plugin>
154             <artifactId>maven-eclipse-plugin</artifactId>
155             <version>2.9</version>
156             <configuration>
157                 <additionalProjectnatures>
158
159                     <projectnature>org.springframework.ide.eclipse.core.springnatu
160                         re</projectnature>
161
162                     </additionalProjectnatures>
163                     <additionalBuildcommands>
164
165                         <buildcommand>org.springframework.ide.eclipse.core.springbuild
166                             er</buildcommand>
167                     </additionalBuildcommands>
168                     <downloadSources>true</downloadSources>
169                     <downloadJavadocs>true</downloadJavadocs>
170                 </configuration>
171             </plugin>
172             <plugin>
173                 <groupId>org.apache.maven.plugins</groupId>
174                 <artifactId>maven-compiler-plugin</artifactId>
175                 <version>2.5.1</version>
176                 <configuration>
177                     <source>1.6</source>
178                     <target>1.6</target>
179                     <compilerArgument>-Xlint:all</compilerArgument>
180                     <showWarnings>true</showWarnings>
181                     <showDeprecation>true</showDeprecation>
182                 </configuration>
183             </plugin>
184             <plugin>
185                 <groupId>org.codehaus.mojo</groupId>
186                 <artifactId>exec-maven-plugin</artifactId>
187                 <version>1.2.1</version>
188                 <configuration>
189                     <mainClass>org.test.intl.Main</mainClass>
190                 </configuration>
191             </plugin>
192         </plugins>
193     </build>
194 </project>

```



```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee"
3      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4      xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
5          http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd">
6      <!-- The definition of the Root Spring Container shared by all Servlets and
7          Filters -->
8      <context-param>
9          <param-name>contextConfigLocation</param-name>
10         <param-value>/WEB-INF/spring/root-context.xml</param-value>
11     </context-param>
12     <!-- Creates the Spring Container shared by all Servlets and Filters -->
13     <listener>
14         <listener-class>org.springframework.web.context.ContextLoaderListener</listen
15         e
16         r-class>
17     </listener>
18     <!-- Processes application requests -->
19     <servlet>
20         <servlet-name>appServlet</servlet-name>
21
22         <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-clas
23         s>
24         <init-param>
25             <param-name>contextConfigLocation</param-name>
26             <param-value>/WEB-INF/spring/appServlet/servlet-context.xml</param-value>
27         </init-param>
28         <load-on-startup>1</load-on-startup>
29     </servlet>
30     <servlet-mapping>
31         <servlet-name>appServlet</servlet-name>
32         <url-pattern>/</url-pattern>
33     </servlet-mapping>
34 </web-app>
```