Web Service “CalculationOfPremium”

**Description**

This service calculate premium for a given set of coverages or for all product’s coverages based on family relationships between insureds and their age.

**There are two forms of using this service.**

The first is using to calculate the premiums for all product coverages. In this case mandatory input parameters are <FamilyStructure>, <BirthDateList> and <State>. The result <PremiumList> is array of 15 real values - one for each coverage.

The second is using to calculate one premium for given set of coverages. In this case mandatory input parameters are <FamilyStructure>, <BirthDateList>, <State> and <Contract>. The result <PremiumList> is array of only one real value which is sum of premiums for all coverages from input parameter <Contract>.

**Format of input parameters**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input Parameter** | **Type** | **Format** | **Possible Values** | **Meaning** |
| <FamilyStructure> | String | String[12] | divorced | Single Parent |
| married | Family |
| partner1 | Couples |
| single3 | Single |
| <BirthDateList>  <BirthDate/>  </BirthDateList> | Date | YYYY-MM-DD |  |  |
| <State> | String | String[3] | ACT | Australian Capital Territory |
| NSW | New South Wales |
| NT | Northern Territory |
| QLD | Queensland |
| SA | South Australia |
| TAS | Tasmania |
| VIC | Victoria |
| WA | Western Australia |
| <Contract>  <ProductOption>  <PackageList>  <Package>  <CoverageList>  <Coverage>  <Identifier>  ……… | Integer |  | 67 | Young Hospital No Excess |
| 52 | Basic Hospital Excess $250 |
| 68 | Basic Hospital Excess $500 |
| 53 | Mid Hospital No Excess |
| 69 | Mid Hospital Excess $250 |
| 70 | Mid Hospital Excess $500 |
| 54 | Top Hospital No Excess |
| 71 | Top Hospital $250 |
| 72 | Top Hospital $500 |
| 73 | Basic Extras 55% back |
| 74 | Basic Extras 70% back |
| 40 | Top Extras 55% back |
| 41 | Top Extras 70% back |
| 42 | Top Extras 85% back |
| 63 | Ultra Health Cover |

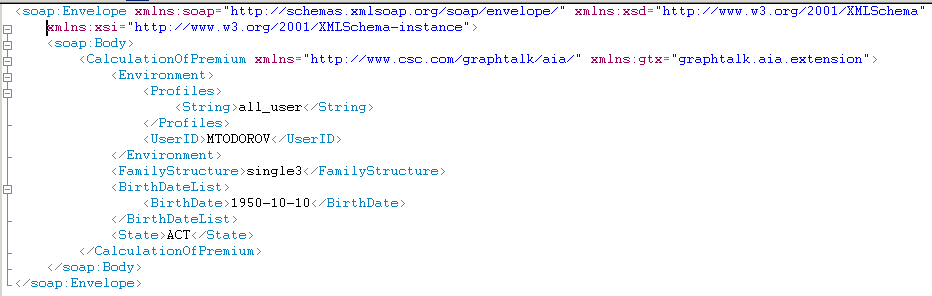
**Format of output parameters**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Output Parameter** | **Type** | **Value** | **Coverage Identifier** | **Coverage Label** |
| <PremiumList>  <Premium>Val\_1</Premium>  ……….  <Premium>Val\_15</Premium>  </PremiumList>  Note: In case of calculating premiums for all coverages the result is array of 15 values. The order of values is important as each value corresponded to certain coverage. | Real | Val\_1 | 67 | Young Hospital No Excess |
| Val\_2 | 52 | Basic Hospital Excess $250 |
| Val\_3 | 68 | Basic Hospital Excess $500 |
| Val\_4 | 53 | Mid Hospital No Excess |
| Val\_5 | 69 | Mid Hospital Excess $250 |
| Val\_6 | 70 | Mid Hospital Excess $500 |
| Val\_7 | 54 | Top Hospital No Excess |
| Val\_8 | 71 | Top Hospital $250 |
| Val\_9 | 72 | Top Hospital $500 |
| Val\_10 | 73 | Basic Extras 55% back |
| Val\_11 | 74 | Basic Extras 70% back |
| Val\_12 | 40 | Top Extras 55% back |
| Val\_13 | 41 | Top Extras 70% back |
| Val\_14 | 42 | Top Extras 85% back |
| Val\_15 | 63 | Ultra Health Cover |
| <PremiumList>  <Premium>Val\_1</Premium>  </PremiumList>  Note: In case of calculating premium for given set of coverages the result is array of only one value corresponded to calculated premium for this set of coverages. | Real | Val\_1 | N/A | N/A |

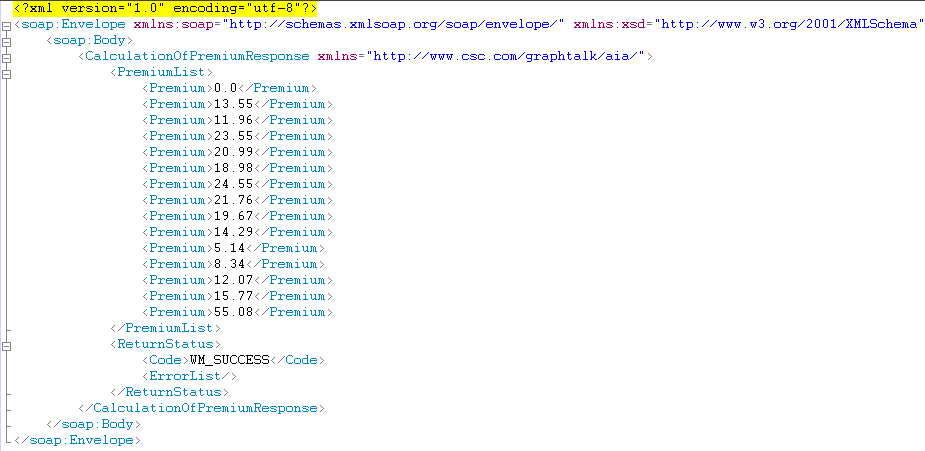
**Examples**

**Request Flow Variant 1.1**

*Calculating premiums for all coverages about single person, born on 10/10/1950 (age 62 years), lives in ACT state*

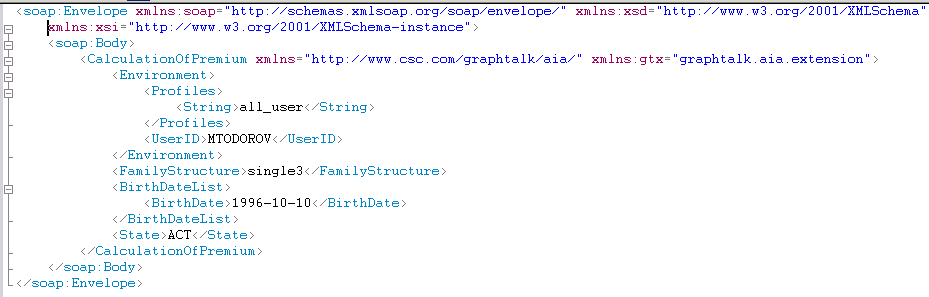


**Response Flow Variant 1.1**

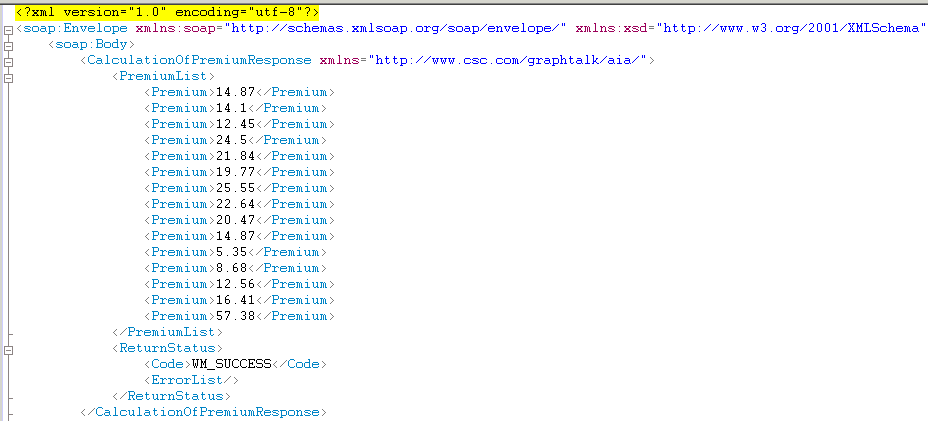


**Request Flow Variant 1.2**

*Calculating premiums for all coverages about single person, born on 10/10/1996 (age 16 years), lives in ACT state*

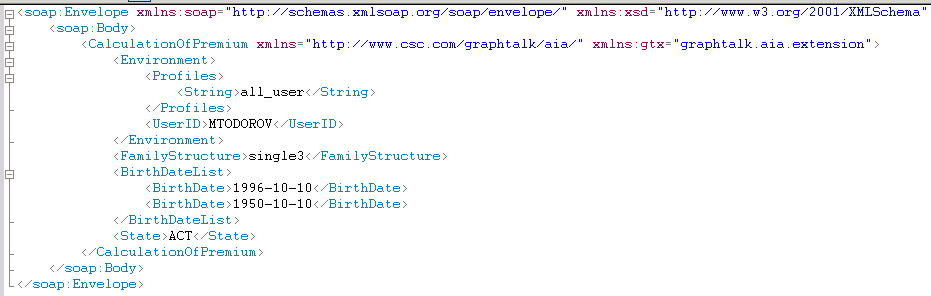


**Response Flow Variant 1.2**



**Request Flow Variant 1.3**

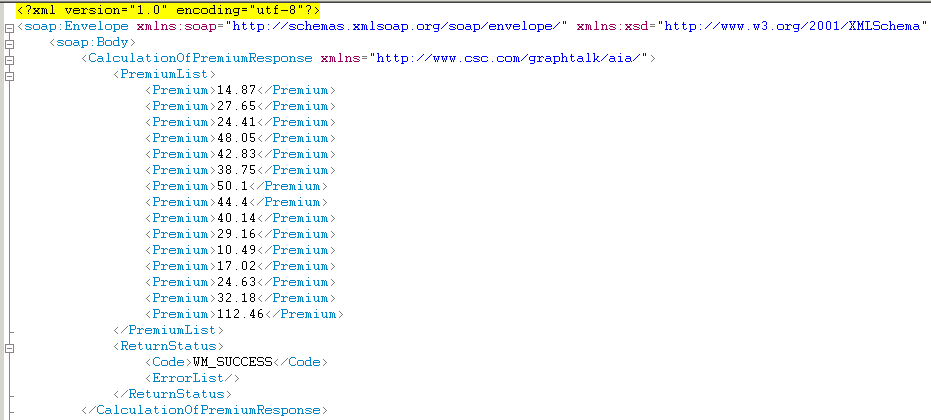
*Calculating premiums for all coverages about two persons which are married, the first born on 10/10/1996 (age 16 years) and the second born on 10/10/1950 (age 62 years), they live in ACT state*



Note that in this example tag <FamilyStructure> contain value “single3” because at this time only this value is available. The value “married” must be used instead “single3”.

**Response Flow Variant 1.3**

*The result is array of premiums for all coverages. Each premium is equal to sum of the corresponded premiums for first and second person. See Response Flow Variant 1.1 and 1.2*



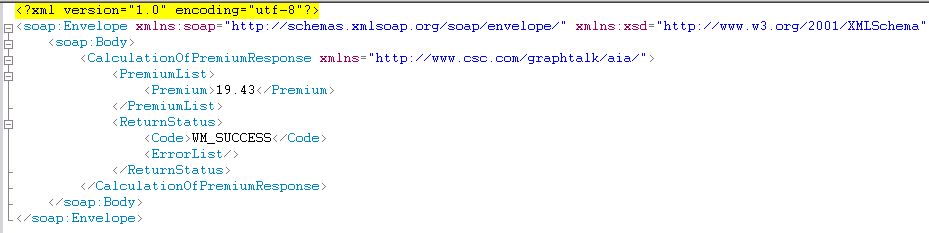
**Request Flow Variant 2.1**

*Calculating premium for given set of coverages about single person, born on 10/10/1950 (age 62 years), lives in ACT state*



**Response Flow Variant 2.1**

*The result is sum of corresponded premiums for coverages with identifiers 73 and 74. See Response Flow Variant 1.1 and Format of Input Parameters table*



**Request Flow Variant 2.2**

*Calculating premium for given set of coverages about single person, born on 10/10/1996 (age 16 years), lives in ACT state*



**Response Flow Variant 2.2**



**Request Flow Variant 2.3**

*Calculating premium for given set of coverages (73 and 74 in example) about two persons which are married, the first born on 10/10/1996 (age 16 years) and the second born on 10/10/1950 (age 62 years), they live in ACT state*



**Response Flow Variant 2.3**

*The result is sum of both premiums for each person. See Response Flow Variant 2.1 and 2.2*



**Request Flow Variant 2.4**

*Calculating premium for given set of coverages where coverage with identifier 68 is the main coverage (*Basic Hospital Excess $500*) which is in one package and coverages with identifiers 73 and 74 (*Basic Extras 55% back, Basic Extras 70% back*)which are extra coverages located in other package.*



**Response Flow Variant 2.4**

****

**XML files contains all of request flows above.**



