འབྲུག་རྒྱལ་འཛིན་གཙུག་ལག་སློབ་སྡེ།།

College of Science and Technology
Rinchending: Bhutan

Continuous Practical Assignment II

Application of OOP Principles

Programming Methodology (BIA101)

Bachelors of Business Intelligence (BBI)

འབྲུག་རྒྱལ་འཛིན་གཙུག་ལག་སློབ་སྡེ།།

College of Science and Technology
Rinchending: Bhutan

RUB Wheel of Academic Law: Academic Dishonesty

Section H2 of the Royal University of Bhutan's *Wheel of Academic Law* provides the following definition of academic dishonesty:

Academic dishonesty may be defined as any attempt by a student to gain an unfair advantage in any assessment. It may be demonstrated by one of the following:

1. **Collusion:** the representation of a piece of unauthorised group work as the work of a single candidate.
2. **Commissioning:** submitting an assignment done by another person as the student's own work.
3. **Duplication**: the inclusion in coursework of material identical or substantially similar to material which has already been submitted for any other assessment within the University.
4. **False declaration**: making a false declaration in order to receive special consideration by an Examination Board or to obtain extensions to deadlines or exemption from work.
5. **Falsification of data**: presentation of data in laboratory reports, projects, etc., based on work purported to have been carried out by the student, which has been invented, altered or copied by the student.
6. **Plagiarism**: the unacknowledged use of another's work as if it were one's own.

Examples are:

- verbatim copying of another's work without acknowledgement.
- paraphrasing of another's work by simply changing a few words or altering the order of presentation, without acknowledgement.
- ideas or intellectual data in any form presented as one's own without acknowledging the source(s).
- making significant use of unattributed digital images such as graphs, tables, photographs, etc. taken from test books, articles, films, plays, handouts, internet, or any other source, whether published or unpublished.
- submission of a piece of work which has previously been assessed for a different award or module or at a different institution as if it were new work.
- use of any material without prior permission of copyright from appropriate authority or owner of the materials used".

Royal University of Bhutan

འབྲུག་རྒྱལ་འཛིན་གཙུག་ལག་སློབ་སྡེ།།

College of Science and Technology
Rinchending: Bhutan

CAP II: Application for OOP Principles

Introduction

The CAP2 for BIA101 involves programming an application using OOP principles to calculate personal income tax(PIT) for individuals.

You are a tax intermediary tasked with calculating personal income tax for individuals; develop an application using python to correctly calculate total tax payable for an individual's income.

Take into consideration the real values and tax brackets when calculating tax.
Refer to department or revenue and customs along with the tax act of bhutan for more information.

Here is some information on personal income tax collected from the department of revenue and customs:

Personal Income Tax (PIT) PIT is a tax levied on the personal income of an individual on an accrual basis from any or more of the following four sources:

(a) Employment Income

- monthly basic salary allowances
- fees & remuneration
- bonus
- commission
- leave encashment
- share of profits received by an employee
- consultancy income by an non licensed consultant and
- other benefits received free or at a concessional rate.

(b) Rental Income Income earned from real properties such as:

- land(s)
- building(s)
- house(s)
- factory building(s)
- warehouse(s) and
- any other structures generating income.

(c) Dividend Income

- Income received from holding of shares in a company registered in Bhutan. However, dividend income does not include dividends received in the form of bonus shares. Dividend income exceeding Nu.30,000 shall be subjected to TDS @ 10% on the whole amount. Dividend income shall be taxed on receipt basis.

(d) Income from Interest

- Exempted from income year 2016.

(e) Income from sale of Cash Crop

- Exempted from income year 2020.

(f) Income from Other Sources

Income from other sources shall be taxed on accrual basis and it's income is derived from any of the following sources:

1. Income from hire of privately owned vehicles, plants and machinery but excluding trucks and taxis;
2. Income from intellectual property rights such as copyrights, patent, trademark, design, model or any artistic work or scientific work.

Specific deductions/exemptions Income Sources Specific Deductions/Exemptions

1. Salary Income : Less PF and GIS contributions.

2. Rental Income : Less 20% on repairs and maintenance,

interest payments, urban taxes and insurance premium.

3. Dividend Income : Less Nu. 30,000 specific exemption and

interest on loans for shareholding.

4. Other Income : Less 30% on Gross Other Income.

General Deductions In addition, the following general deductions are given

1. Education allowance up to a max of Nu. 350,000 per child.
2. Life insurance premium.
3. Self education allowance up to a max Nu. 350,000 per taxpayer.
4. Donations up to a max of 5% of the total adjusted gross income.
5. Sponsored children education expense up to max of Nu. 350,000 per child.

Sl No. :	Income Slab	Tax Rate
1	Up to Nu. 300,000	0%
2	Nu.300,001 to Nu. 400,000	10%
3	Nu. 400,001 to Nu. 650,000	15%
4	Nu. 650,001 to Nu. 1,000,000	20%
5	Nu. 1,000,001 to Nu. 1,500,000	25%
6	Nu. 1,500,001 and up	30%
Surcharge at the rate of 10% shall be levied on (PIT)	Personal Income Tax, if the annual Personal Income Tax	is equal to or more than Nu. 1,000,000.

Your application should include the following:

1. **Exception Handling**: Implement limitations to exempt individuals earning less than the minimum taxable income from paying taxes.
2. **OOP Principles**: Utilise classes and objects to structure the code, leveraging Object-Oriented Programming principles effectively.
3. **Intelligible Comments**: Ensure all comments within the code are meaningful and comprehensible.
4. **Assumptions**: The code should accommodate the following assumptions:
    a. Assume individuals have only one source of income, namely Salary.
    b. Incorporate functionality to discern between Contract and Regular employee positions within organisations. Note: Government organisations typically do not offer pension(PF) schemes to contract employees, whereas certain corporations and private entities do.
    c. Integrate functionality to account for different types of organisations where individuals are employed, such as Government, Private, and Corporate sectors.
    d. Deductibles:
        - NPPF: Deductible for Pension Scheme/Provident Fund (PF).
        - Tax deductions for children, whether they are in school or not.
        - GIS: Deductible for Group Insurance Scheme.

Submission

1. Create a **Public** github repository with the name "YourEnrollmentNumber_BIA101_CAPII". Example: 02908728_BIA101_CAPII
2. Push all work done for CAPII, i.e, source code to your repo before the deadline.
3. Submit the link to your repository on google classroom.
4. The submission should be made in the section "CAP2 BIA101A" for Section A students and "CAP2 BIA101B" for Section B students.
5. Submit the link in the Google Classroom submission portal.
6. ***SUBMISSION DEADLINE: 10th May 11.59 PM (Late submissions will not be entertained, the submission portal will close on the deadline.)***

NOTE:

- **PLAGIARISM WILL NOT BE TOLERATED - DO NOT SHARE CODE IF FOUND BOTH (OR ALL) STUDENTS WILL BE GIVEN ZERO (0).**

Resources to help with the Assignment:

- Python OOP online tutorials
- Python OOP online course

Marking Rubrics: 100 Marks converted to 15 Marks

Criteria	Description	Weightage	Accomplished (4)	Good (3)	Satisfactory (2)	Needs Improvement (1)
Code readability	Structured, documented, formatted	20%	Modular components, descriptive names, commented logic, properly formatted	Mostly meaningful naming and comments, proper formatting	Limited structure, minimal comments, minor formatting issues	No modularization, cryptic names, no comments, formatting issues
Functionality	Completeness of required gameplay features	30%	All features flawlessly implemented, no bugs	Complete with minor gaps, few bugs	Major features work, some gaps/bugs	Major gaps, bugs prevent functionality
OOP Principles	Encapsulation, inheritance, abstraction	25%	Excellent class abstraction and encapsulation, advanced patterns	Appropriate encapsulation and inheritance	Weak abstraction and OO design	No custom classes or OO principles
Error Handling	Anticipating and handling errors	25%	All edge cases handled elegantly with custom messaging	Reasonable validation and failure handling, few unhandled errors	Basic handling of some errors	Little to no error handling

Royal University of Bhutan

འབྲུག་རྒྱལ་འཛིན་གཙུག་ལག་སློབ་སྡེ།།

College of Science and Technology
Rinchending: Bhutan