SRD

Équipe CREEi

avr. 12, 2022

Menu:

Le Simulateur de revenu disponible (SRD) a été mis au point par l'équipe de la Chaire de recherche sur les enjeux économiques intergénérationnels, une chaire conjointe HEC Montréal-ESG UQAM. Il permet, pour tout ménage, de calculer le revenu disponible ainsi que les impôts payés, les déductions, les crédits d'impôts et les principaux transferts obtenus, autant au fédéral qu'au provincial, pour les années 2016 à 2020 (le Québec et l'Ontario sont modélisés pour le moment ; il est possible d'utiliser aisément l'une de ces deux structures fiscales pour toute autre province). Pour les années 2020 et 2021, les principales mesures d'urgence liées à la COVID-19 ont également été intégrées : PCU, PCUE, PIRTE, PCRE, majorations du crédit de TPS/TVH, de l'Allocation canadienne pour enfants et du programme de la sécurité de la vieillesse (SRG et SV).

Par rapport aux autres outils existants, le SRD offre une très grande flexibilité et une transparence inégalée. En effet, tout ménage – quelles que soient sa composition et ses caractéristiques socio-économiques – peut être simulé dans le SRD, ce qui permet à l'utilisateur de simuler la base de données de son choix. De plus, le code du simulateur est public et modifiable, ce qui permet d'évaluer les effets de différents scénarios et de modifier les valeurs des paramètres et la structure du simulateur.

Le SRD est écrit en langage Python, un langage simple, rapide et moderne. Afin de pouvoir l'utiliser, il faut s'assurer d'avoir au préalable installé une distribution à jour de Python, par exemple à l'aide d'Anaconda. Dans tous les cas, les exigences minimales pour utiliser le SRD sont Python 3.6+ avec numpy, pandas et xlrd. Bien que ce ne soit pas essentiel, il sera également utile de se familiariser un minimum, au préalable, avec le fonctionnement des environnements Python et avec le vocabulaire utilisé dans la présente documentation (p.ex. fonction, classe, instance, profil).

Pour rester informé.e des mises à jour du SRD (environ 1 à 2 fois par an), inscrivez-vous à notre liste d'envoi dédiée.

CHAPITRE 1

Premiers pas avec le SRD

1.1 Installation du SRD

On peut installer facilement le SRD en utilisant pip :

```
pip install srd
```

Le SRD dépend d'un autre package, le SRPP, qui permet de simuler les revenus de pensions publiques (RPC et RRQ) et les cotisations à ces régimes. Ce package est installé automatiquement en installant le SRD.

Dans un notebook ou un projet, on invoque le SRD en ajoutant la commande suivante :

```
import srd
```

Pour désinstaller le SRD et le SRPP, il suffit d'utiliser pip :

```
pip uninstall srd srpp
```

1.1.1 Installation alternative

Si l'utilisation de pip n'est pas possible, un fichier .zip avec le SRD et sa dépendance, le SRPP, peut être téléchargé de Github (choisissez le fichier *srd_with_dep.zip* dans le dernier « release »). Pour utiliser le SRD, il suffit d'extraire les fichiers du .zip dans le répertoire de son choix et de travailler de ce dernier ou d'ajouter ce répertoire au chemin d'accès (par exemple en utilisant le module *sys*). Un tutoriel a également été inclus dans le fichier .zip. Si l'on ne veut plus le SRD, il suffit d'effacer les répertoires *srd* et *srpp*.

Le SRD est fourni « tel quel », sous une licence MIT.

En cas de questions, commentaires ou suggestions, n'hésitez pas à nous contacter (*Personne-contact*).

CHAPITRE 2

Les acteurs

Les fonctions d'acteurs permettent de déclarer les profils.

2.1 Personnes

class srd.**Person**(*age=50*, *male=True*, *earn=0*, *rpp=0*, *cpp=0*, *net_cap_gains=0*, *prev_cap_losses=0*, *cap_gains_exempt=0*, *othtax=0*, *othntax=0*, *inc_rrsp=0*, *self_earn=0*, *div_elig=0*, *div_other_can=0*, *con_rrsp=0*, *con_non_rrsp=0*, *con_rpp=0*, *union_dues=0*, *donation=0*, *gift=0*, *years_can=None*, *disabled=False*, *widow=False*, *med_exp=0*, *ndays_chcare_k1=0*, *ndays_chcare_k2=0*, *asset=0*, *oas_years_post=0*, *months_cerb_cesb=0*, *student=False*, *essential_worker=False*, *emp_temp_constraints=False*, *hours_month=None*, *prev_inc_work=None*, *dep_senior=False*, *home_support_cost=0*, *months_ei=0*, *months_crb=0*, *pub_drug_insurance=False*, *tax_shield=False*)

Classe pour définir une personne.

Ceci définit une personne et son profil en termes de revenus et d'actifs.

Paramètres

- **age** (*int*) – âge de l'individu
- **male** (*bool*) – prend la valeur True si l'individu est un homme
- **earn** (*float*) – revenu de travail
- **rpp** (*float*) – revenu de régime complémentaire de retraite (RCR)
- **cpp** (*float*) – revenu du Régime de rentes du Québec (RRQ) ou du Régime de pensions du Canada (RPC)
- **net_cap_gains** (*float*) – gains (ou pertes si valeur négative) nets en capital réalisés dans l'année
- **prev_cap_losses** (*float*) – pertes en capital nettes d'autres années (avec facteur d'inclusion partielle déjà appliqué)
- **cap_gains_exempt** (*float*) – exonération des gains en capital admissibles demandée (sur gains en capital nets) ; soumis à un plafond à vie
- **othtax** (*float*) – autre revenu imposable
- **othntax** (*float*) – autre revenu non-imposable
- **inc_rrsp** (*float*) – revenu de REER (retrait de fonds)

— **self_earn** (*float*) – revenu de travail autonome
— **div_elig** (*float*) – montant réel des dividendes déterminés (canadiens)
— **div_other_can** (*float*) – montant réel des dividendes ordinaires (canadiens)
— **con_rrsp** (*float*) – cotisation REER
— **con_non_rrsp** (*float*) – cotisation autre que REER (p.ex. à un CELI ou à des comptes non enregistrés)
— **con_rpp** (*float*) – cotisation à un régime de pension agréé (RPA)
— **union_dues** (*float*) – cotisations syndicales, professionnelles ou autres
— **donation** (*float*) – don de bienfaisance et autres dons
— **gift** (*float*) – dons de biens culturels et écosensibles
— **years_can** (*int*) – nombre d'années vécues au Canada depuis l'âge de 18 ans lorsque la Pension de la sécurité de la vieillesse (PSV) est demandée
— **disabled** (*boolean*) – statut d'invalidité selon les règles fiscales (le même statut est utilisé pour l'ensemble des mesures fiscales modélisées)
— **widow** (*boolean*) – statut de veuf/veuve selon le Programme de la sécurité de la vieillesse
— **med_exp** (*float*) – montant des dépenses en santé admissibles selon les règles fiscales (le même montant est utilisé pour tous les paliers de gouvernement)
— **ndays_chcare_k1** (*float*) – nombre de jours de garde du premier enfant
— **ndays_chcare_k2** (*float*) – nombre de jours de garde du second enfant
— **asset** (*float*) – valeur marchande des actifs (avoirs liquides) comptabilisés aux fins d'admissibilité à l'aide sociale (vérifier la définition selon la province)
— **oas_years_post** (*int*) – nombre d'années de report pour la PSV (après 65 ans)
— **months_cerb_cesb** (*int*) – nombre de mois pour lesquels la PCU ou la PCUE est demandée
— **student** (*boolean*) – statut d'étudiant aux fins d'admissibilité à la PCUE
— **essential_worker** (*boolean*) – True si travailleur essentiel (au Québec seulement)
— **hours_month** (*float*) – nombre d'heures travaillées par mois
— **prev_inc_work** (*float*) – revenu du travail de l'année précédente
— **dep_senior** (*boolean*) – True si la personne aînée n'est pas autonome aux du crédit d'impôt pour maintien à domicile des aînés
— **home_support_cost** (*float*) – dépenses engagées pour des services de maintien à domicile rendus ou à être rendus à partir du jour du 70e anniversaire
— **months_ei** (*int*) – nombre de mois pour lesquels l'Assurance-Emploi sont demandée
— **months_crb** (*int*) – nombre de mois pour lesquels la PCRE est demandée
— **pub_drug_insurance** (*boolean*) – True si la personne doit cotiser à l'Assurance médicaments du Québec (pas d'assurance médicaments privée)
— **tax_shield** (*boolean*) – True si la personne demande le crédit d'impôt bouclier ficscal au Québec

Fonctions incluses dans cette classe :
(Cliquez pour afficher les détails)

- attach_prev_work_inc()

srd.Person.**attach_prev_work_inc**(*self*, *prev_work_inc*)

Fonction qui ajoute le revenu du travail de l'an passé s'il est disponible, ou l'approxime avec le revenu du travail de l'année courante sinon.

Paramètres **prev_work_inc** (*float*) – revenu de travail de l'année précédente

- attach_inc_work_month()

srd.Person.**attach_inc_work_month**(*self*, *earn*, *self_earn*)

Fonction qui convertit le revenu de travail annuel en revenu mensuel et vice versa.

On entre le revenu de travail annuel ou mensuel (liste avec 12 éléments) et le revenu de travail annuel et mensuel deviennent des attributs de la personne.

Paramètres
— **earn** (*float or list*) – revenu de travail salarié
— **self_earn** (*float or list*) – revenu de travail autonome

- inc_work()

srd.Person.**inc_work**()

Fonction qui retourne le revenu de travail.

Inclut le revenu de travail autonome.

Renvoie Revenu de travail.

Type renvoyé float

- inc_non_work()

srd.Person.**inc_non_work**()

Fonction qui retourne le total des revenus autres que les revenus du travail.

Renvoie Revenu provenant de sources autres que le travail.

Type renvoyé float

- inc_tot()

srd.Person.**inc_tot**()

Fonction qui retourne le revenu total.

Ce revenu total contient les montants réels des dividendes de sociétés canadiennes (et non les montants imposables).

Renvoie Revenu total.

Type renvoyé float

- compute_months_cerb_cesb()

srd.Person.**compute_months_cerb_cesb**(*self*, *months_cerb_cesb*, *student*)

Fonction qui établit le nombre de mois de PCU ou de PCUE selon le nombre de mois pour lesquels la personne demande la prestation et selon son statut d'étudiant.

Paramètres
— **months_cerb_cesb** (*int*) – nombre de mois pour lesquels la prestation est demandée
— **student** (*boolean*) – True si la personne est étudiante (ou l'était en décembre 2019)

- copy()

srd.Person.**copy**(*self*)

Fonction qui produit une copie des attributs de la personne.

- reset()

srd.Person.**reset**(*self*)

Fonction qui utilise la copie des attributs de la personne pour réinitialiser l'instance de la personne.

2.2 Dépendants (enfant ou proche à charge)

class srd.**Dependent**(*age*, *disa=False*, *child_care=0*, *school=None*, *home_care=None*, *med_exp=0*)

Classe pour définir un dépendant.

Ceci définit un dépendant et son profil.

Paramètres

- **age** (*int*) – âge de l'individu
- **disa** (*boolean*) – statut d'invalidité
- **child_care** (*float*) – dépenses réelles de frais de garde
- **school** (*float*) – dépenses de scolarité
- **home_care** (*float*) – dépenses engagées pour des services de maintien à domicile rendus ou à être rendus à partir du jour du 70e anniversaire
- **med_exp** (*float*) – dépenses en santé admissibles

2.3 Ménages

class srd.**Hhold**(*first*, *second=None*, *prov='qc'*, *n_adults_in_hh=None*, *prev_fam_net_inc_prov=None*)

Classe pour définir un ménage.

Ceci définit un ménage et son profil.

Paramètres

- **first** (Person) – instance Person du 1er membre du couple
- **second** (Person) – instance Person du 2e membre du couple, s'il y a lieu
- **prov** (*str*) – province (qc = Québec)
- **n_adults_in_hh** (*int*) – nombre d'adultes (18 ans et plus) dans le ménage

Fonctions incluses dans cette classe - les fonctions relatives au revenu utilisent des quantités déjà calculées au niveau individuel par la classe principale du calculateur (*tax*), soit le calculateur de revenu disponible :
(Cliquez pour afficher les détails)

- adjust_n_adults()

srd.Hhold.**adjust_n_adults**(*self*, *n_adults_in_hh*)

Fonction qui calcule le nombre d'adultes dans le ménage si celui-ci n'est pas fourni.

Paramètres **n_adults_in_hh** (*float*) – nombre d'adultes dans le ménage s'il est fourni, None sinon

Renvoie Nombre d'adultes dans le ménage.

Type renvoyé float

- fam_inc_work()

srd.Hhold.**fam_inc_work**()

Fonction qui calcule le revenu de travail du ménage.

Renvoie Revenu de travail du ménage.

Type renvoyé float

- fam_inc_non_work()

srd.Hhold.**fam_inc_non_work**(*self*)

Fonction qui calcule le revenu familial de sources autres que le travail.

Renvoie Revenu familial de sources autres que le travail.

Type renvoyé float

- fam_inc_tot()

srd.Hhold.**fam_inc_tot**()

Fonction qui calcule le revenu familial total.

Renvoie Revenu familial total.

Type renvoyé float

- fam_after_tax_inc()

srd.Hhold.**fam_after_tax_inc**()

Fonction qui calcule le revenu familial après impôts.

Renvoie Revenu familial après impôts.

Type renvoyé float

- fam_disp_inc()

srd.Hhold.**fam_disp_inc**()

Fonction qui additionne les revenus disponibles du conjoint pour obtenir le revenu disponible familial.
Il s'agit du revenu disponible après impôts, cotisations sociales, épargne (positive ou négative) et prestations.

Renvoie Revenu familial disponible, après impôts et cotisations.

Type renvoyé float

- child_care_exp()

srd.Hhold.**child_care_exp**()

Fonction qui calcule la dépense en frais de garde pour le ménage.

Renvoie Montant total des dépenses de frais de garde.

Type renvoyé float

- fam_net_inc_prov()

srd.Hhold.**fam_net_inc_prov**()

Fonction qui calcule le revenu familial net pour l'impôt et les programmes provinciaux.

Renvoie Revenu familial net provincial.

Type renvoyé float

- fam_net_inc_fed()

srd.Hhold.**fam_net_inc_fed**()

Fonction qui calcule le revenu familial net pour l'impôt et les programmes fédéraux.

Renvoie Revenu familial net fédéral.

Type renvoyé float

- add_dependent()

srd.Hhold.**add_dependent**(*self*, **dependents*)

Fonction pour ajouter un ou plusieurs dépendant(s).

Paramètres **dependent** (Dependent) – instance de la classe Dependent ou liste d'instances de la classe Dependent

- count()

srd.Hhold.**count**(*self*)

Fonction pour calculer le nombre d'enfants dans différentes catégories d'âge.

- compute_max_split()

srd.Hhold.**compute_max_split**(*self*)

Fonction qui calcule le montant maximal de revenu de pension pouvant être fractionné, et qui l'attache à chaque conjoint du ménage dans l'attribut *max_split*.

- assess_elig_split()

srd.Hhold.**assess_elig_split**(*self*)

Fonction qui établit si le ménage est admissible pour le fractionnement du revenu de pension, et qui l'attache au ménage dans l'attribut *elig_split*.

- copy()

srd.Hhold.**copy**(*self*)

Fonction qui produit une copie des attributs du ménage et des personnes dans le ménage.

- reset()

srd.Hhold.**reset**(*self*)

Fonction qui utilise la copie des attributs du ménage et des personnes pour réinitialiser l'instance du ménage.

CHAPITRE 3

Impôt fédéral

3.1 Survol

La fiscalité fédérale est prise en charge par le module *federal*. Ce module contient un gabarit que nous documentons ci-dessous ainsi que des classes dérivées spécifiques pour chaque année.

La fonction *form* permet de choisir l'année de la déclaration de revenus et crée une instance de déclaration pour cette année. L'instance est retournée par la fonction suivante :

`srd.federal.`**`form`**`(`*year*`)`

Fonction qui permet de sélectionner le formulaire d'impôt fédéral par année.

Renvoie Une instance du formulaire pour l'année sélectionnée.

Type renvoyé class instance

Voici les mesures fiscales fédérales prises en compte dans ce module :

- L'impôt des particuliers
- L'abattement du Québec à l'impôt fédéral
- Les déductions et crédits d'impôt pour les cotisations sociales (RRQ/RPC, AE et RQAP)
- Les déductions pour la perte en capital nette d'autres années et pour gain en capital exonéré
- La déduction fédérale pour frais de garde
- Le crédit d'impôt non-remboursable pour dividendes
- Le crédit d'impôt non-remboursable en raison de l'âge
- Le crédit d'impôt non-remboursable pour revenu de retraite
- Le crédit d'impôt non-remboursable pour invalidité
- Le crédit d'impôt non-remboursable pour dons de bienfaisance
- Le crédit d'impôt non-remboursable pour personne à charge admissible
- Le montant canadien pour emploi
- Le montant pour personne à charge admissible
- Le montant pour époux ou conjoint de fait
- Le transfert de crédits non-remboursables d'un conjoint à l'autre
- Les crédits d'impôt pour frais médicaux (remboursable et non-remboursable)
- Le crédit remboursable pour la taxe sur les produits et services/taxe de vente harmonisée (TPS/TVH)
- L'allocation canadienne pour les travailleurs (ACT) et son supplément pour invalidité

— L'allocation canadienne pour enfants (ACE) et son supplément pour jeunes enfants (ACESJE)

La liste exhaustive des éléments calculés dans le module est présentée dans la section ci-dessous.

3.2 Gabarit de déclaration

Nous utilisons un gabarit afin de créer les déclarations chaque année. Quand l'impôt change seulement au niveau des paramètres d'une année à l'autre, la déclaration ira seulement chercher les nouveaux paramètres. Quand des fonctions changent, l'utilisateur n'a qu'à modifier les fonctions touchées (ou à en ajouter de nouvelles). Toutes les modifications de fonction survenues après l'année 2016 sont indiquées dans la section suivante.

srd.federal.**template**()

Gabarit pour l'impôt fédéral.

Nous présentons ici la liste des fonctions incluses dans le gabarit :
(Cliquez sur le nom pour afficher les détails)

- file()

srd.federal.form_2016.**file**(*self*, *hh*)

Fonction qui permet de calculer l'impôt.

Cette fonction est celle qui calcule les déductions,les crédits non-remboursables et remboursables et l'impôt net.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- calc_gross_income()

srd.federal.form_2016.**calc_gross_income**(*self*, *p*)

Fonction qui calcule le revenu total (brut).

Cette fonction correspond au revenu total d'une personne aux fins de l'impôt.

Paramètres **p** (Person) – instance de la classe Person

- calc_net_income()

srd.federal.form_2016.**calc_net_income**(*self*, *p*)

Fonction qui calcule le revenu net au sens de l'impôt.

Paramètres **p** (Person) – instance de la classe Person

- calc_taxable_income()

srd.federal.form_2016.**calc_taxable_income**(*self*, *p*)

Fonction qui calcule le revenu imposable au sens de l'impôt.

Paramètres **p** (Person) – instance de la classe Person

- calc_deduc_gross_income()

srd.federal.form_2016.**calc_deduc_gross_income**(*self*, *p*, *hh*)

Fonction qui calcule les déductions.

Cette fonction fait la somme des différentes déductions du contribuable.

Paramètres

— **p** (Person) – instance de la classe Person

— **hh** (Hhold) – instance de la classe Hhold

- chcare()

srd.federal.form_2016.**chcare**(*self*, *p*, *hh*)

Fonction qui calcule la déduction fédérale pour frais de garde.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie

Montant de la déduction pour frais de garde.

Cette fonction calcule le montant reçu en fonction des frais de garde, de l'âge des enfants et du revenu le moins élevé du couple. Le montant est reçu par le conjoint qui a le revenu le moins élevé.

Type renvoyé float

- cpp_deduction()

srd.federal.form_2016.**cpp_deduction**(*self*, *p*)

Fonction qui calcule la déduction pour les cotisations au RRQ/RPC pour les travailleurs autonomes.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant de la déduction.

Type renvoyé float

- qpip_deduction()

srd.federal.form_2016.**qpip_deduction**(*self*, *p*)

Fonction qui calcule la déduction pour les cotisations au RQAP pour les travailleurs autonomes.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant de la déduction.

Type renvoyé float

- calc_deduc_net_income()

srd.federal.form_2016.**calc_deduc_net_income**(*self*, *p*)

Fonction qui calcule les déductions suivantes : 1. Pertes en capital nettes d'autres années ; 2. Déduction pour gain en capital exonéré.

Permet une déduction maximale égale aux gains en capital taxables nets.

Paramètres **p** (Person) – instance de la classe Person

- calc_tax()

srd.federal.form_2016.**calc_tax**(*self*, *p*)

Fonction qui calcule l'impôt à payer selon la table d'impôt.

Cette fonction utilise la table d'impôt de l'année en cours.

Paramètres **p** (Person) – instance de la classe Person

- calc_non_refundable_tax_credits()

srd.federal.form_2016.**calc_non_refundable_tax_credits**(*self*, *p*, *hh*)

Fonction qui calcule les crédits d'impôt non-remboursables.

Cette fonction fait la somme de tous les crédits modélisés en appelant les fonctions définies ci-après.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

- get_age_cred()

srd.federal.form_2016.**get_age_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable en raison de l'âge.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- get_cpp_contrib_cred()

srd.federal.form_2016.**get_cpp_contrib_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable pour cotisations au RRQ/RPC.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- get_qpip_cred()

srd.federal.form_2016.**get_qpip_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable pour cotisations au RQAP de travailleur salarié.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- get_qpip_self_cred()

srd.federal.form_2016.**get_qpip_self_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable pour cotisations au RQAP de travailleur autonome.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- get_empl_cred()

srd.federal.form_2016.**get_empl_cred**(*self*, *p*)

Fonction qui calcule le montant canadien pour emploi.
Ce crédit est non-remboursable.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- get_pension_cred()

srd.federal.form_2016.**get_pension_cred**(*self*, *p*, *hh*)

Fonction qui calcule le crédit d'impôt non-remboursable pour revenu de retraite.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la class Hhold

Renvoie Montant du crédit.

Type renvoyé float

- get_disabled_cred()

srd.federal.form_2016.**get_disabled_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable pour invalidité.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- get_med_exp_nr_cred()

srd.federal.form_2016.**get_med_exp_nr_cred**(*self*, *p*, *hh*)

Fonction qui calcule le crédit d'impôt non-remboursable pour frais médicaux.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

- get_donations_cred()

srd.federal.form_2016.**get_donations_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable pour dons.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- div_tax_credit()

srd.federal.form_2016.**div_tax_credit**(*self*, *p*)

Fonction qui calcule le crédit d'impôt pour dividendes.

Paramètres **p** (Person) – instance de la classe Person

- calc_refundable_tax_credits()

srd.federal.form_2016.**calc_refundable_tax_credits**(*self*, *p*, *hh*)

Fonction qui fait la somme des crédits remboursables, en appelant les fonctions suivantes, décrites ailleurs dans cette page : *abatment*, *ccb*, *get_witb*, *get_witbds*, *med_exp*, *gst_hst_credit*.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

- abatment()

srd.federal.form_2016.**abatment**(*self*, *p*, *hh*)

Fonction qui calcule l'abattement du Québec à l'impôt fédéral.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de l'abattement.

Type renvoyé float

- ccb()

srd.federal.form_2016.**ccb**(*self*, *p*, *hh*, *iclaw=True*)

Fonction qui calcule l'Allocation canadienne pour enfants (ACE).

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold
- **iclaw** (*boolean*) – récupération des prestations si True ; pas de récupération si False

Renvoie Montant de l'ACE.

Type renvoyé float

- get_witb()

srd.federal.form_2016.**get_witb**(*self*, *p*, *hh*)

Fonction qui calcule l'Allocation canadienne pour les travailleurs (ACT).

Avant 2019, celle-ci était appelée la Prestation fiscale pour le revenu de travail (PFRT).

Dans le cas d'un couple, la prestation est répartie au prorata des revenus de travail.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de l'ACT.

Type renvoyé float

- get_witbds()

srd.federal.form_2016.**get_witbds**(*self*, *p*, *hh*)

Fonction qui calcule le supplément pour invalidité à la Prestation fiscale pour le revenu de travail (SIPFRT).

À partir de 2019, le SIPFRT devient le supplément pour invalidité à l'Allocation canadienne pour les travailleurs (ACT).

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du supplépment pour invalidité.

Type renvoyé float

- compute_witb_witbds()

srd.federal.form_2016.**compute_witb_witbds**(*self*, *p*, *hh*, *rate*, *base*, *witb_max*, *claw_rate*, *exemption*)

Fonction appelée par *get_witb* et *get_witbds* pour calculer le montant de la PFRT / l'ACT et de son supplément.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold
- **rate** (*float*) – taux appliqué au revenu du travail
- **base** (*float*) – montant de base de la PFRT / l'ACT
- **witb_max** (*float*) – montant maximal de la PFRT / l'ACT
- **claw_rate** – taux de réduction
- **exemption** (*float*) – exemption

Renvoie Montant de la PFRT / l'ACT ou du supplément.

Type renvoyé float

- med_exp()

srd.federal.form_2016.**med_exp**(*self*, *p*, *hh*)

Fonction qui calcule le crédit remboursable pour frais médicaux.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

- gst_hst_credit()

srd.federal.form_2016.**gst_hst_credit**(*self*, *p*, *hh*)

Fonction qui calcule le crédit pour la taxe sur les produits et services/taxe de vente harmonisée (TPS/TVH).
Le montant du crédit est reçu par le conjoint au revenu imposable le plus élevé.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

- get_spouses_cred()

srd.federal.form_2016.**get_spouses_cred**(*self*, *p*, *hh*)

Fonction qui calcule le montant pour époux ou conjoint de fait.
Ce crédit est non-remboursable.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

- get_dep_cred()

srd.federal.form_2016.**get_dep_cred**(*self*, *p*, *hh*)

Fonction qui calcule le montant pour personne à charge admissible.
Ce crédit est non-remboursable.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

- get_spouse_transfer()

srd.federal.form_2016.**get_spouse_transfer**(*self*, *p*, *hh*)

Fonction qui récupère le surplus des crédits non-remboursables transférables au conjoint (s'il y lieu).

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

- get_ei_contrib_cred()

srd.federal.form_2016.**get_ei_contrib_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable pour cotisations à l'assurance emploi.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

Le gabarit collige les résultats dans un formulaire d'impôt qui sera rattaché à la personne sous la forme d'un dictionnaire Python. Cette procédure permet de différencier les variables générées par l'impôt des attributs d'une personne qui font partie de son profil. C'est la fonction *create_return()* qui fait ce travail.

srd.federal.**create_return**()

3.3 Fonctions spécifiques ou modifiées

2016

class srd.federal.**form_2016**

Formulaire d'impôt de 2016.

2017

class srd.federal.**form_2017**

Formulaire d'impôt de 2017.

2018

class srd.federal.**form_2018**

Formulaire d'impôt de 2018.

2019

class srd.federal.**form_2019**

Formulaire d'impôt de 2019.

- cpp_deduction()

srd.federal.form_2019.**cpp_deduction**(*self*, *p*)

Fonction qui calcule la déduction pour les cotisations au RRQ/RPC de base pour les travailleurs autonomes et pour le régime supplémentaire du RRQ/RPC pour tous.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant de la déduction.

Type renvoyé float

2020

class srd.federal.**form_2020**

Formulaire d'impôt de 2020.

- compute_basic_amount()

srd.federal.form_2020.**compute_basic_amount**(*self*, *p*)

Fonction qui calcule le montant personnel de base.
Le calcul de ce montant change en 2020.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant personnel de base.

Type renvoyé float

- calc_net_income()

srd.federal.form_2020.**calc_net_income**(*self*, *p*)

Fonction qui calcule le revenu net au sens de l'impôt.

Paramètres **p** (Person) – instance de la classe Person

- repayments_ei()

srd.federal.form_2020.**repayments_ei**(*self*, *p*)

Fonction qui calcule le montant du remboursement de prestations d'assurance-emploi et qui ajuste le montant des prestations, le revenu net et le revenu brut.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du remboursement.

Type renvoyé float

- calc_refundable_tax_credits()

srd.federal.form_2020.**calc_refundable_tax_credits**(*self*, *p*, *hh*)

Fonction qui fait la somme des crédits remboursables, en appelant les fonctions suivantes, décrites ailleurs dans cette page : *abatment*, *ccb*, *get_witb*, *get_witbds*, *med_exp*, *gst_hst_credit*.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

- oas_gis_covid_bonus()

srd.federal.form_2020.**oas_gis_covid_bonus**(*self*, *p*)

Fonction qui calcule le montant supplémentaire unique de Pension de sécurité de vieillesse (PSV) et de Supplément de revenu garanti (SRG). Pour l'année 2020, le gouvernement a versé une prestation unique non imposable de 300$ aux bénéficiaires de la PSV. Les bénéficiaires du SRG ont aussi eu droit à un montant additionnel de 200$.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du paiement unique de SV et de SRG.

Type renvoyé float

2021

class srd.federal.**form_2021**

Formulaire d'impôt de 2021.

- calc_refundable_tax_credits()

srd.federal.form_2021.**calc_refundable_tax_credits**(*self*, *p*, *hh*)

Fonction qui fait la somme des crédits remboursables, en appelant les fonctions suivantes, décrites ailleurs dans cette page : *abatment*, *ccb*, *get_witb*, *get_witbds*, *med_exp*, *gst_hst_credit*.

Paramètres

- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

- ccb()

srd.federal.form_2021.**ccb**(*self*, *p*, *hh*, *iclaw=True*)

Fonction qui calcule pour 2021 l'Allocation canadienne pour enfants (ACE) avec l'ACE supplément pour jeunes enfants (ACESJE), soit de moins de 6 ans.

Paramètres

- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold
- **iclaw** (*boolean*) – récupération des prestations si True ; pas de récupération si False

Renvoie Montant de l'ACE.

Type renvoyé float

CHAPITRE 4

Impôt du Québec

4.1 Survol

L'impôt du Québec est pris en charge par le module *quebec*. Ce module contient un gabarit que nous documentons ci-dessous ainsi que des classes dérivées spécifiques pour chaque année.

La fonction *form* permet de choisir l'année de la déclaration de revenus et ira tirer une instance de la déclaration pour cette année. L'instance est retournée par la fonction.

`srd.quebec.form`(*year*)

Fonction qui permet de sélectionner le formulaire d'impôt provincial par année.

Paramètres `year` (*int*) – année (présentement entre 2016 et 2021)

Renvoie Une instance du formulaire pour l'année sélectionnée.

Type renvoyé class instance

Les mesures fiscales provinciales prises en compte dans ce module sont les suivantes :

- L'impôt des particuliers
- Les déductions pour les cotisations sociales (RRQ/RPC et RQAP)
- La déduction pour travailleur
- Les déductions pour pertes en capital nettes d'autres années et pour gain en capital exonéré
- Le crédit d'impôt non-remboursable pour dividendes
- Le crédit d'impôt non-remboursable en raison de l'âge, pour personne vivant seule et pour revenu de retraite
- Le crédit d'impôt non-remboursable pour la prolongation de carrière
- Le crédit d'impôt non-remboursable pour cotisations syndicales et professionnelles
- Le crédit d'impôt non-remboursable pour dons
- Le crédit d'impôt non-remboursable pour invalidité
- Le transfert de crédits non-remboursables d'un conjoint à l'autre
- Les crédits d'impôt pour frais médicaux (remboursable et non-remboursable)
- Le crédit d'impôt remboursable pour solidarité
- Le crédit d'impôt remboursable pour maintien à domicile des aînés
- Le crédit d'impôt remboursable pour soutien aux ainés
- Le crédit d'impôt remboursable pour frais de garde d'enfant
- Le crédit d'impôt remboursable accordant une allocation aux familles

— Le crédit d'impôt remboursable relatif à la prime au travail
— La cotisation au régime d'assurance médicaments du Québec
— La cotisation au Fonds des services de santé (FSS)
— La contribution santé (2016)
— La contribution additionnelle pour les services de garde éducatifs à l'enfance subventionnés (2016 à 2018)
— Les crédits d'impôt remboursable attribuant une prestation exceptionnelle et un montant ponctuel pour pallier la hausse du coût de la vie
— Le crédit d'impôt bouclier fiscal

La liste exhaustive des éléments calculés dans le module est présentée dans la section ci-dessous.

4.2 Gabarit de déclaration

Nous utilisons un gabarit afin de créer les déclarations chaque année. Quand l'impôt change seulement au niveau des paramètres d'une année à l'autre, la déclaration ira seulement chercher les nouveaux paramètres. Quand des fonctions changent, l'utilisateur n'a qu'à modifier les fonctions touchées (ou à en ajouter de nouvelles). Toutes les modifications de fonction survenues après l'année 2016 sont indiquées ci-dessous.

srd.quebec.**template**()

Gabarit pour l'impôt provincial québécois.

Nous reproduisons ici la spécification du gabarit. Il est basé sur la déclaration de 2016.
(Cliquez sur le nom pour afficher les détails)

- file()

srd.quebec.form_2020.**file**(*self*, *hh*)

Fonction qui permet de calculer l'impôt.

Cette fonction est celle qui calcule les déductions, les crédits non-remboursables et remboursables et l'impôt net.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- calc_gross_income()

srd.quebec.form_2020.**calc_gross_income**(*self*, *p*)

Fonction qui calcule le revenu total (brut).

Cette fonction correspond au revenu total d'une personne aux fins de l'impôt.

Paramètres **p** (Person) – instance de la classe Person

- calc_net_income()

srd.quebec.form_2020.**calc_net_income**(*self*, *p*)

Fonction qui calcule le revenu net au sens de l'impôt.

Cette fonction correspond au revenu net d'une personne aux fins de l'impôt. On y soustrait les déductions.

Paramètres **p** (Person) – instance de la classe Person

- calc_taxable_income()

srd.quebec.form_2020.**calc_taxable_income**(*self*, *p*)

Fonction qui calcule le revenu imposable au sens de l'impôt.

Paramètres **p** (Person) – instance de la classe Person

- calc_deduc_gross_income()

srd.quebec.form_2020.**calc_deduc_gross_income**(*self*, *p*)

Fonction qui calcule les déductions.

Cette fonction fait la somme des différentes déductions du contribuable.

Paramètres **p** (Person) – instance de la classe Person

- work_deduc()

srd.quebec.form_2020.**work_deduc**(*self*, *p*)

Fonction qui calcule la déduction pour travailleur.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant de la déduction.

Type renvoyé float

- cpp_qpip_deduction()

srd.quebec.form_2020.**cpp_qpip_deduction**(*self*, *p*)

Déduction pour les cotisations RRQ/RPC et au RQAP pour le travail autonome.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant de la déduction.

Type renvoyé float

- calc_deduc_net_income()

srd.quebec.form_2020.**calc_deduc_net_income**(*self*, *p*)

Fonction qui calcule les déductions suivantes : 1. Pertes en capital nettes d'autres années ; 2. Déduction pour gain en capital exonéré.

Permet une déduction maximale égale aux gains en capital taxables nets.

Paramètres **p** (Person) – instance de la classe Person

- calc_tax()

srd.quebec.form_2020.**calc_tax**(*self*, *p*)

Fonction qui calcule l'impôt à payer selon la table d'impôt.

Cette fonction utilise la table d'impôt de l'année en cours.

Paramètres **p** (Person) – instance de la classe Person

- calc_non_refundable_tax_credits()

srd.quebec.form_2020.**calc_non_refundable_tax_credits**(*self*, *p*, *hh*)

Fonction qui calcule les crédits d'impôt non-remboursables.

Cette fonction fait la somme de tous les crédits d'impôt modélisés.

Paramètres

— **p** (Person) – instance de la classe Person

— **hh** (Hhold) – instance de la classe Hhold

- get_age_cred()

srd.quebec.form_2020.**get_age_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable en raison de l'âge.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- get_living_alone_cred()

srd.quebec.form_2020.**get_living_alone_cred**(*self*, *p*, *hh*)

Fonction qui calcule le crédit d'impôt non-remboursable pour personne vivant seule.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

- get_pension_cred()

srd.quebec.form_2020.**get_pension_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable pour revenu de retraite.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- get_nrtcred_clawback()

srd.quebec.form_2020.**get_nrtcred_clawback**(*self*, *p*, *hh*)

Fonction qui calcule la récupération de la somme des crédits non-remboursables 1. en raison de l'âge ; 2. pour personne vivant seule ; et 3. pour revenu de retraite.

Cette fonction utilise le revenu net du ménage.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de la récupération.

Type renvoyé float

- get_exp_worker_cred()

srd.quebec.form_2020.**get_exp_worker_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable pour les travailleurs d'expérience. Depuis 2019, renommé crédit d'impôt pour la prolongation de carrière.

On fait l'hypothèse que les travailleurs de 65 ans sont nés le 1er janvier de l'année en cours. (En réalité, les revenus gagnés avant et après le 65e anniversaire sont soumis à des traitements différents, ce qui complique beaucoup le modèle mais change peu les résultats.)

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- get_donations_cred()

srd.quebec.form_2020.**get_donations_cred**(*self*, *p*)

Fonction qui remplace dans le gabarit (classe *srd.quebec.template*) la fonction du même nom, et calcule le crédit d'impôt non-remboursable pour dons.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit

Type renvoyé float

- get_union_dues_cred()

srd.quebec.form_2020.**get_union_dues_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable pour cotisations syndicales et professionnelles.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- get_disabled_cred()

srd.quebec.form_2020.**get_disabled_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt pour invalidité.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- get_med_exp_cred()

srd.quebec.form_2020.**get_med_exp_cred**(*self*, *p*, *hh*)

Fonction qui calcule le crédit d'impôt non-remboursable pour frais médicaux.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

- div_tax_credit()

srd.quebec.form_2020.**div_tax_credit**(*self*, *p*)

Fonction qui calcule le crédit d'impôt pour dividendes.

Paramètres **p** (Person) – instance de la classe Person

- calc_refundable_tax_credits()

srd.quebec.form_2020.**calc_refundable_tax_credits**(*self*, *p*, *hh*)

Fonction qui fait la somme des crédits remboursables.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

- chcare()

srd.quebec.form_2020.**chcare**(*self*, *p*, *hh*)

Fonction qui calcule le crédit d'impôt remboursable pour frais de garde.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie

Montant du crédit pour frais de garde.

Cette fonction calcule le montant reçu en fonction du nombre d'enfants, de la situation familiale (couple/monoparental) et du revenu.

Type renvoyé float

- witb()

srd.quebec.form_2020.**witb**(*self*, *p*, *hh*)

Fonction qui calcule la prime au travail.

Le calcul est fait en tenant compte du revenu de travail, du revenu du ménage et de la présence d'un enfant à charge. Pour les couples, la prime est partagée au prorata des revenus de travail.

Notes : le supplément à la prime au travail et la prime au travail adaptée ne sont pas calculés.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de la prime au travail par individu.

Type renvoyé float

- home_support()

srd.quebec.form_2020.**home_support**(*self*, *p*, *hh*)

Fonction qui calcule le crédit d'impôt pour maintien à domicile des aînés.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

- med_exp()

srd.quebec.form_2020.**med_exp**(*self*, *p*, *hh*)

Fonction qui calcule le crédit remboursable pour frais médicaux.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

- ccap()

srd.quebec.form_2020.**ccap**(*self*, *p*, *hh*)

Fonction qui calcule le crédit d'impôt remboursable accordant une allocation aux familles (CIRAAF) (qui s'appelait le Soutien aux enfants avant 2019). Seules les composantes suivantes sont modélisées : l'allocation famille et le supplément pour l'achat de fournitures scolaires.

Cette fonction calcule le montant reçu selon le nombre d'enfants, la situation familiale (couple/monoparental) et le revenu.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit d'impôt remboursable accordant une allocation aux familles (CIRAAF).

Type renvoyé float

- calc_contributions()

srd.quebec.form_2020.**calc_contributions**(*self*, *p*, *hh*)

Fonction qui remplace la fonction antérieure du même nom, et calcule les contributions.

Cette fonction fait la somme des contributions du contribuable. La contribution additionnelle pour service de garde éducatifs à l'enfance subventionnés est abolie en 2019.

Paramètres
- **p** (Person) – instance de la classe Person

— **hh** (Hhold) – instance de la classe Hhold

- health_contrib()

srd.quebec.form_2020.**health_contrib**(*self*, *p*, *hh*)

Fonction qui calcule la contribution santé.

Cette fonction calcule le montant dû en fonction du revenu net. La contribution santé a été abolie en 2017.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

- add_contrib_subsid_chcare()

srd.quebec.form_2020.**add_contrib_subsid_chcare**(*self*, *p*, *hh*)

Contribution additionnelle pour les services de garde éducatifs à l'enfance subventionnés.

Cette fonction calcule le montant dû en fonction du nombre de jours de garde et du revenu familial. Chaque conjoint paie en fonction du nombre de jours de garde sur son relevé 30. La contribution a été abolie en 2019.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

- solidarity()

srd.quebec.form_2020.**solidarity**(*self*, *p*, *hh*)

Fonction qui calcule le crédit d'impôt pour solidarité.

Cette fonction calcule le montant reçu par chacun des conjoints en fonction du revenu familial de l'année fiscale courante (en réalité, le calcul est basé sur le revenu de l'année précédente).

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

- drug_insurance_contrib()

srd.quebec.form_2020.**drug_insurance_contrib**(*self*, *hh*)

Fonction qui sert à calculer la cotisation au régime d'assurance médicaments du Québec.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- contrib_hsf()

srd.quebec.form_2020.**contrib_hsf**(*self*, *p*)

Fonction qui calcule la cotisation au Fonds des services de santé (FSS).

Paramètres **p** (Person) – instance de la classe Person

- get_spouse_transfer()

srd.quebec.form_2020.**get_spouse_transfer**(*self*, *p*, *hh*)

Fonction qui récupère le surplus des crédits non rembousables tranferables au conjoint (s'il y lieu).

Paramètres **p** (Person) – instance de la classe Person

- cost_of_living()

srd.quebec.form_2020.**cost_of_living**(*self*, *p*, *hh*)

Fonction qui calcule le crédit d'impôt remboursable attribuant une prestation exceptionnelle pour pallier la hausse du coût de la vie et le crédit d'impôt remboursable attribuant un montant ponctuel pour pallier la hausse du coût de la vie.

En vigueur en 2021 seulement.

p : Person instance de la classe Person

hh : Hhold instance de la classe Hhold

- tax_shield()

srd.quebec.form_2020.**tax_shield**(*self*, *p*, *hh*)

Fonction qui calcule le crédit d'impôt Bouclier fiscal.

La part de ce crédit liée à la prime au travail est partagée proportionnellement au revenu des conjoints par rapport au revenu famillial.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Le gabarit collige les résultats dans un formulaire d'impôt qui sera rattaché à la personne sous la forme d'un dictionnaire Python. Cette procédure permet de différencier les attributs d'une personne qui font partie de son profil des variables générées par l'impôt. C'est la fonction *create_return()* qui fait ce travail.

srd.quebec.**create_return**()

4.3 Fonctions spécifiques ou modifiées par année

2016

class srd.quebec.**form_2016**

Formulaire d'impôt de 2016.

2017

class srd.quebec.**form_2017**

Formulaire d'impôt de 2017.

- calc_contributions()

srd.quebec.form_2017.**calc_contributions**(*self*, *p*, *hh*)

Fonction qui remplace dans le gabarit (classe *srd.quebec.template*) la fonction du même nom, et calcule les contributions.

Cette fonction fait la somme des contributions du contribuable. La contribution santé est abolie en 2017.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

- get_donations_cred()

srd.quebec.form_2017.**get_donations_cred**(*self*, *p*)

Fonction qui remplace dans le gabarit (classe *srd.quebec.template*) la fonction du même nom, et calcule le crédit d'impôt non-remboursable pour dons.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit

Type renvoyé float

- ccap()

srd.quebec.form_2017.**ccap**(*self*, *p*, *hh*)

Fonction qui calcule le crédit d'impôt remboursable accordant une allocation aux familles (CIRAAF) (qui s'appelait le Soutien aux enfants avant 2019). Seules les composantes suivantes sont modélisées : l'allocation famille et le supplément pour l'achat de fournitures scolaires.

Cette fonction calcule le montant reçu selon le nombre d'enfants, la situation familiale (couple/monoparental) et le revenu.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit d'impôt remboursable accordant une allocation aux familles (CIRAAF).

Type renvoyé float

2018

class srd.quebec.**form_2018**

Formulaire d'impôt de 2018.

- senior_assist()

srd.quebec.form_2018.**senior_assist**(*self*, *p*, *hh*)

Fonction qui remplace dans le gabarit (classe *srd.quebec.template*) la fonction du même nom, et calcule le crédit remboursable pour soutien aux ainés. En vigueur à partir de 2018.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

2019

class srd.quebec.**form_2019**

Formulaire d'impôt de 2019.

- cpp_qpip_deduction()

srd.quebec.form_2019.**cpp_qpip_deduction**(*self*, *p*)

Déduction pour les cotisations RRQ/RPC et au RQAP pour le travail autonome.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant de la déduction.

Type renvoyé float

- calc_contributions()

srd.quebec.form_2019.**calc_contributions**(*self*, *p*, *hh*)

Fonction qui remplace la fonction antérieure du même nom, et calcule les contributions.

Cette fonction fait la somme des contributions du contribuable. La contribution additionnelle pour service de garde éducatifs à l'enfance subventionnés est abolie en 2019.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

2020

class srd.quebec.**form_2020**

Formulaire d'impôt de 2020.

- calc_deduc_gross_income()

srd.quebec.form_2020.**calc_deduc_gross_income**(*self*, *p*)

Fonction qui calcule les déductions.

Cette fonction fait la somme des différentes déductions du contribuable.

Paramètres **p** (Person) – instance de la classe Person

2021

class srd.quebec.**form_2021**

Formulaire d'impôt de 2021.

- calc_deduc_gross_income()

srd.quebec.form_2021.**calc_deduc_gross_income**(*self*, *p*)

Fonction qui calcule les déductions.

Cette fonction fait la somme des différentes déductions du contribuable.

Paramètres **p** (Person) – instance de la classe Person

- cost_of_living()

srd.quebec.form_2021.**cost_of_living**(*self*, *p*, *hh*)

Fonction qui calcule les crédits d'impôt remboursables attribuant une prestation exceptionnelle et un montant ponctuel pour pallier la hausse du coût de la vie.

Paramètres

— **p** (Person) – instance de la classe Person

— **hh** (Hhold) – instance de la classe Hhold

CHAPITRE 5

Impôt de l'Ontario

5.1 Survol

L'impôt de l'Ontario est pris en charge par le module *ontario*. Ce module contient un gabarit que nous documentons ci-dessous ainsi que des classes dérivées spécifiques pour chaque année.

La fonction *form* permet de choisir l'année de la déclaration de revenus et ira tirer une instance du rapport pour cette année. L'instance est retournée par la fonction.

srd.ontario.**form**(*year*)

Fonction qui permet de sélectionner le formulaire d'impôt provincial par année.

Paramètres **year** (*int*) – année (présentement entre 2016 et 2021)

Renvoie Une instance du formulaire pour l'année sélectionnée.

Type renvoyé class instance

Les principales mesures fiscales provinciales prises en compte dans ce module sont les suivantes :

- L'impôt des particuliers
- La surtaxe de l'Ontario
- La réduction de l'impôt de l'Ontario
- Le crédit d'impôt non-remboursable provincial en raison de l'âge et pour revenu de retraite
- Le crédit d'impôt non-remboursable pour cotisations au RRQ/RPC
- Le crédit d'impôt non-remboursable pour invalidité
- Le crédit d'impôt non-remboursable pour frais médicaux
- Le crédit d'impôt non-remboursable pour dons de l'Ontario
- Le crédit d'impôt non-remboursable pour les personnes et les familles à faible revenu
- Le crédit d'impôt non-remboursable pour dividendes
- Le montant pour époux ou conjoint de fait
- Le crédit d'impôt remboursable de taxe de vente de l'Ontario
- L'Allocation ontarienne pour enfants
- La contribution santé de l'Ontario

La liste exhaustive des éléments calculés dans le module est présentée dans la section ci-dessous.

5.2 Gabarit du rapport

Nous utilisons un gabarit afin de créer les rapports chaque année. Quand l'impôt change seulement au niveau des paramètres d'une année à l'autre, le rapport ira seulement chercher les nouveaux paramètres. Quand des fonctions changent, l'utilisateur n'a qu'à modifier les fonctions touchées (ou à en ajouter de nouvelles). Toutes les modifications de fonction survenues après l'année 2016 sont indiquées ci-dessous.

srd.ontario.**template**()

Gabarit pour l'impôt provincial ontarien.

Nous reproduisons ici la spécification du gabarit. Il est basé sur le rapport de 2016.
(Cliquez sur le nom pour afficher les détails)

- file()

srd.ontario.form_2016.**file**(*self*, *hh*)

Fonction qui permet de calculer l'impôt.

Cette fonction est celle qui calcule les déductions, les crédits non-remboursables et remboursables et l'impôt net.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- copy_fed_return()

srd.ontario.form_2016.**copy_fed_return**(*self*, *p*)

Fonction qui copie le revenu brut, les déductions, ainsi que les revenus net et imposable du formulaire fédéral.

Paramètres **p** (Person) – instance de la classe Person

- calc_tax()

srd.ontario.form_2016.**calc_tax**(*self*, *p*)

Fonction qui calcule l'impôt à payer selon la table d'impôt.

Cette fonction utilise la table d'impôt de l'année en cours.

Paramètres **p** (Person) – instance de la classe Person

- calc_non_refundable_tax_credits()

srd.ontario.form_2016.**calc_non_refundable_tax_credits**(*self*, *p*, *hh*)

Fonction qui fait la somme de tous les crédits d'impôt non-remboursables modélisés.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

- get_age_cred()

srd.ontario.form_2016.**get_age_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable provincial en raison de l'âge.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit

Type renvoyé float

- get_spouse_cred()

srd.ontario.form_2016.**get_spouse_cred**(*self*, *p*, *hh*)

Fonction qui calcule le montant pour époux ou conjoint de fait.
Ce crédit est non-remboursable.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

- get_cpp_contrib_cred()

srd.ontario.form_2016.**get_cpp_contrib_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable pour cotisations au RRQ/RPC.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- get_pension_cred()

srd.ontario.form_2016.**get_pension_cred**(*self*, *p*, *hh*)

Fonction qui calcule le crédit d'impôt non-remboursable pour revenu de retraite.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la class Hhold

Renvoie Montant du crédit.

Type renvoyé float

- get_disabled_cred()

srd.ontario.form_2016.**get_disabled_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable pour invalidité.
Seule la portion pour le contribuable majeur lui-même est modélisée.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- get_med_exp_cred()

srd.ontario.form_2016.**get_med_exp_cred**(*self*, *p*, *hh*)

Fonction qui calcule le crédit d'impôt non-remboursable pour frais médicaux.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

- get_donations_cred()

srd.ontario.form_2016.**get_donations_cred**(*self*, *p*)

Fonction qui calcule le crédit d'impôt non-remboursable pour dons de l'Ontario.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant du crédit.

Type renvoyé float

- surtax()

srd.ontario.form_2016.**surtax**(*self*, *p*)

Fonction qui calcule la surtaxe de l'Ontario.

Paramètres **p** (Person) – instance de la classe Person

- div_tax_credit()

srd.ontario.form_2016.**div_tax_credit**(*self*, *p*)

Fonction qui calcule le crédit d'impôt pour dividendes de l'Ontario.

Paramètres **p** (Person) – instance de la classe Person

- tax_reduction()

srd.ontario.form_2016.**tax_reduction**(*self*, *p*, *hh*)

Fonction qui calcule la réduction de l'impôt de l'Ontario.

Ce montant est non-remboursable.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

- calc_refundable_tax_credits()

srd.ontario.form_2016.**calc_refundable_tax_credits**(*self*, *p*, *hh*)

Fonction qui fait la somme des crédits remboursables.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

- ocb()

srd.ontario.form_2016.**ocb**(*self*, *p*, *hh*)

Fonction qui calcule l'Allocation ontarienne pour enfants.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de l'Allocation ontarienne pour enfants.

Type renvoyé float

- ostc()

srd.ontario.form_2016.**ostc**(*self*, *p*, *hh*)

Crédit de taxe de vente de l'Ontario.

Ce crédit est remboursable.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant du crédit.

Type renvoyé float

- calc_contributions()

srd.ontario.form_2016.**calc_contributions**(*self*, *p*)

Fonction qui fait la somme des contributions du contribuable (actuellement, seule la contribution santé est incluse).

Paramètres **p** (Person) – instance de la classe Person

- health_contrib()

srd.ontario.form_2016.**health_contrib**(*self*, *p*)

Contribution santé de l'Ontario (Ontario health premium).

Cette fonction calcule le montant dû en fonction du revenu imposable.

Paramètres **p** (Person) – instance de la classe Person

Le gabarit collige les résultats dans un formulaire d'impôt qui sera rattaché à la personne sous la forme d'un dictionnaire Python. Cette procédure permet de différencier les attributs d'une personne qui font partie de son profil des variables générées par l'impôt. C'est la fonction *create_return()* qui fait ce travail.

srd.ontario.**create_return**()

5.3 Fonctions spécifiques ou modifiées par année

2016

class srd.ontario.**form_2016**

Formulaire d'impôt de 2016.

2017

class srd.ontario.**form_2017**

Formulaire d'impôt de 2017.

2018

class srd.ontario.**form_2018**

Formulaire d'impôt de 2018.

2019

class srd.ontario.**form_2019**

Formulaire d'impôt de 2019.

- lift_credit()

srd.ontario.form_2020.**lift_credit**(*self*, *p*, *hh*)

Crédit d'impôt pour les personnes et les familles à faible revenu (Low-income individuals and families tax credit : LIFT).

Ce crédit entre en vigueur en 2019. Il est non-remboursable.

Paramètres

— **p** (Person) – instance de la classe Person

— **hh** (Hhold) – instance de la classe Hhold

2020

class srd.ontario.**form_2020**

Formulaire d'impôt de 2020.

2021

class srd.ontario.**form_2021**

Formulaire d'impôt de 2021.

CHAPITRE 6

Pension de la sécurité de la vieillesse

6.1 Survol

Le programme de la Sécurité à la vieillesse est pris en charge par le module *oas* et comprend la Pension de la sécurité à la vieillesse (PSV), le Supplément de revenu garanti (SRG), l'Allocation (au conjoint) et l'Allocation au survivant. Ce module contient un gabarit que nous documentons ci-dessous ainsi que des classes dérivées spécifiques pour chaque année.

La fonction *program* permet de choisir l'année du programme et ira tirer une instance du programme pour cette année. L'instance est retournée par la fonction.

srd.oas.**program**(*year*, *federal*)

Fonction qui permet de sélectionner le programme par année.

Paramètres

- **year** (*int*) – année (présentement entre 2016 et 2021)
- **federal** (*{srd.federal.form_2016, ..., srd.federal.form_2021}*) – instance de la classe srd.federal.form_xxxx (pour l'année xxxx) du module *federal*

Renvoie Une instance de la classe de l'année sélectionnée.

Type renvoyé class instance

6.2 Gabarit du programme

Nous utilisons un gabarit afin de créer les programmes chaque année. Quand le programme change seulement au niveau des paramètres d'une année à l'autre, nous irons chercher seulement les nouveaux paramètres. Quand des fonctions changent, l'utilisateur n'a qu'à modifier les fonctions touchées (ou à en ajouter de nouvelles). L'avantage des classes dérivées est de ne pas avoir à répéter toutes les fonctions d'une année à l'autre si celles-ci n'ont pas changé.

srd.oas.**template**()

Classe qui contient un gabarit du programme de la Sécurité de la vieillesse (PSV, SRG, Allocation et Allocation au survivant), tel qu'il existait en 2016.

Nous reproduisons ici la spécification du gabarit. Il est basé sur le programme en vigueur en 2016.
(Cliquez sur le nom pour afficher les détails)

- file()

srd.oas.template.template.**file**(*self*, *hh*)

Fonction pour faire une demande au programme et recevoir une prestation, en faisant appel à toutes les fonctions qui suivent.

Ceci calcule les prestations pour la PSV, le SRG, l'Allocation et l'Allocation au survivant.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- eligibility()

srd.oas.template.template.**eligibility**(*self*, *p*, *hh*)

Fonction qui évalue l'admissibilité de la personne à chacun des 4 volets du programme.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

- compute_net_income()

srd.oas.template.template.**compute_net_income**(*self*, *p*, *hh*)

Fonction qui calcule le revenu net (sans la PSV).

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

- compute_net_inc_exemption()

srd.oas.template.template.**compute_net_inc_exemption**(*self*, *hh*)

Fonction qui calcule le revenu en sus de l'exemption aux fins du SRG sur les revenus du travail salarié.

Paramètres **hh** (Hhold) – instance de la classe Hhold

Renvoie Revenu en sus de l'exemption sur les revenus du travail aux fins du calcul du SRG.

Type renvoyé float

- compute_pension()

srd.oas.template.template.**compute_pension**(*self*, *p*, *hh*)

Fonction qui calcule la prestation de PSV.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de la PSV.

Type renvoyé float

- pension_clawback()

srd.oas.template.template.**pension_clawback**(*self*, *p*, *hh*)

Fonction qui calcule la récupération de la PSV, basée sur le revenu net qui inclut la PSV.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de la récupération de la PSV.

Type renvoyé float

- gis()

srd.oas.template.template.**gis**(*self*, *p*, *hh*, *income*, *low_high*)

Fonction qui calcule la prestation de Supplément de revenu garanti.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold
- **income** (*float*) – revenu aux fins du calcul de la récupération du SRG
- **low_high** (*string*) – "low"/"high" pour calcul du bonus de SRG pour très faible revenu

Renvoie Montant du SRG (après récupération).

Type renvoyé float

- survivor_allowance()

srd.oas.template.template.**survivor_allowance**(*self*, *p*, *hh*)

Fonction qui calcule la prestation effective d'Allocation au survivant, incluant la récupération.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Prestation effective d'Allocation au survivant, incluant bonus et récupération.

Type renvoyé float

- couple_allowance()

srd.oas.template.template.**couple_allowance**(*self*, *p*, *hh*)

Fonction qui calcule la prestation effective d'Allocation au conjoint, en tenant compte du revenu.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Prestation effective d'Allocation au conjoint, ajustée en fonction du revenu.

Type renvoyé float

- compute_allowance()

srd.oas.template.template.**compute_allowance**(*self*, *p*, *hh*, *supp_max*)

Fonction qui calcule le montant maximal de l'Allocation au survivant ou de l'Allocation au conjoint.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold
- **supp_max** (*float*) – prestation maximale de SRG

Renvoie Montant maximal de l'Allocation.

Type renvoyé float

6.3 Fonctions spécifiques ou modifiées par année

2016

class srd.oas.programs.**program_2016**(*federal*)

Version du programme de 2016.

Paramètres **federal** (srd.federal.form_2016) – instance de la classe srd.federal.form_2016 du module *federal*

2017

class srd.oas.programs.**program_2017**(*federal*)

Version du programme de 2017.

Paramètres **federal** (srd.federal.form_2017) – instance de la classe srd.federal.form_2017 du module *federal*

2018

class srd.oas.programs.**program_2018**(*federal*)

Version du programme de 2018.

Paramètres **federal** (srd.federal.form_2018) – instance de la classe srd.federal.form_2018 du module *federal*

2019

class srd.oas.programs.**program_2019**(*federal*)

Version du programme de 2019.

Paramètres **federal** (srd.federal.form_2019) – instance de la classe srd.federal.form_2019 du module *federal*

2020

class srd.oas.programs.**program_2020**(*federal*)

Version du programme de 2020.

Paramètres **federal** (srd.federal.form_2020) – instance de la classe srd.federal.form_2020 du module *federal*

- compute_net_inc_exemption()

srd.oas.programs.program_2020.**compute_net_inc_exemption**(*self*, *hh*)

Fonction qui remplace dans le gabarit (classe *srd.oas.template*) la fonction du même nom, et calcule le revenu net incluant l'exemption sur les revenus du travail salarié.

À partir de 2020-2021, les revenus de travail autonome bénéficient également de l'exemption. Les revenus du travail entre 5 000 $ et 10 000 $ bénéficient d'une nouvelle exemption partielle de 50%.

Paramètres **hh** (Hhold) – instance de la classe Hhold

Renvoie Revenu net de l'exemption sur les revenus du travail.

Type renvoyé float

2021

class srd.oas.programs.**program_2021**(*federal*)

Version du programme de 2021.

Paramètres **federal** (srd.federal.form_2021) – instance de la classe srd.federal.form_2021 du module *federal*

- compute_pension()

srd.oas.programs.program_2021.**compute_pension**(*self*, *p*, *hh*)

Fonction qui calcule la prestation de PSV.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de la PSV.

Type renvoyé float

- gis()

srd.oas.programs.program_2021.**gis**(*self*, *p*, *hh*, *income*, *low_high*)

Fonction qui calcule la prestation de Supplément de revenu garanti.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold
— **income** (*float*) – revenu aux fins du calcul de la récupération du SRG
— **low_high** (*string*) – "low"/"high" pour calcul du bonus de SRG pour très faible revenu

Renvoie Montant du SRG (après récupération).

Type renvoyé float

CHAPITRE 7

Cotisations sociales

Les cotisations sociales sont prises en charge par la classe *payroll*. Cette classe prend en charge les cotisations à l'assurance-emploi, au Régime québécois d'assurance parentale (RQAP), au Régime de rentes du Québec (RRQ) et au Régime de pensions du Canada (RPC). Pour les deux dernières, le module *srpp* de la chaire est utilisé et une fonction de la classe *payroll* (*get_cpp_contrib*) ira chercher les cotisations à l'intérieur de ce module.

La classe *payroll* attachera à chaque membre d'un couple un rapport de cotisations qui contiendra toutes les sommes cotisées. Pour le RRQ et le RPC, les cotisations aux régimes de base et celles aux régimes supplémentaires découlant de l'expansion débutée en 2019 sont séparées, car leur traitement fiscal est différent.

class srd.**payroll**(*year*)

Calcul des cotisations sociales : à l'assurance-emploi, au RQAP (Québec), au RRQ (Québec) ainsi qu'au RPC (provinces autres que le Québec).

Paramètres **year** (*int*) – année pour le calcul

compute(*hh*)

Fonction qui appelle *ei*, *qpip* (voir ci-après) et la fonction *get_cpp_contrib* afin de calculer respectivement les cotisations à l'assurance-emploi, à l'assurance parentale (RQAP) et au RRQ/RPC.

Paramètres **hh** (Hhold) – instance de la classe Hhold

get_cpp_contrib(*p*, *hh*)

Fonction pour le calcul des cotisations au RPC et au RRQ, qui appelle le module *srpp*.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montants des cotisations aux régimes de base et supplémentaire du RRQ et du RPC.

Type renvoyé liste de floats

7.1 Assurance emploi

7.1.1 Survol

Le programme d'assurance-emploi est pris en charge par le module *ei*. Ce module contient un gabarit que nous documentons ci-dessous ainsi que des classes dérivées spécifiques pour chaque année.

La fonction *program* permet de choisir l'année du programme et ira tirer une instance du programme pour cette année. L'instance est retournée par la fonction.

srd.ei.**program**(*year*)

Fonction qui permet de sélectionner le programme par année.

Paramètres **year** (*int*) – année (présentement entre 2016 et 2021)

Renvoie Une instance de la classe de l'année sélectionnée.

Type renvoyé class instance

Pour le moment, le programme permet uniquement de calculer les cotisations.

7.1.2 Gabarit du programme

Nous utilisons un gabarit afin de créer les programmes chaque année. Quand le programme change seulement au niveau des paramètres d'une année à l'autre, nous irons chercher seulement les nouveaux paramètres. Quand des fonctions changent, l'utilisateur n'a qu'à modifier les fonctions touchées (ou à en ajouter de nouvelles). L'avantage des classes dérivées est de ne pas avoir à répéter toutes les fonctions d'une année à l'autre si celles-ci n'ont pas changé.

Nous reproduisons ici la spécification du gabarit. Il est basé sur le programme en vigueur en 2016.

class srd.ei.**template**

Programme d'assurance-emploi.

Ce gabarit sert pour l'instant à calculer les cotisations à l'assurance-emploi.

contrib(*p*, *hh*)

Fonction pour calculer les cotisations à l'assurance-emploi.

Paramètres

- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de la cotisation à l'assurance-emploi (annuelle).

Type renvoyé float

7.1.3 Fonctions spécifiques ou modifiées par année

class srd.ei.programs.**program_2016**

Version du programme de 2016.

class srd.ei.programs.**program_2017**

Version du programme de 2017.

class srd.ei.programs.**program_2018**

Version du programme de 2018.

class srd.ei.programs.**program_2019**

Version du programme de 2019.

class srd.ei.programs.**program_2020**

Version du programme de 2020.

class srd.ei.programs.**program_2021**

Version du programme de 2021 (excluant les paramètres en lien avec la COVID-19).

- compute_benefits_covid()

srd.ei.programs.program_2021.**compute_benefits_covid**(*self*, *p*, *hh*)

Fonction pour calculer les prestations de l'assurance emploi qui remplaceraient la PCU (contrefactuel).

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie montant de la prestation

Type renvoyé float

7.2 Régime québécois d'assurance parentale

7.2.1 Survol

Le Régime québécois d'assurance parentale existant au Québec est pris en charge par le module *qpip*. Ce module contient un gabarit que nous documentons ci-dessous, ainsi que des classes dérivées spécifiques pour chaque année.

La fonction *program* permet de choisir l'année du programme et ira tirer une instance du programme pour cette année. L'instance est retournée par la fonction.

srd.qpip.**program**(*year*)

Fonction qui permet de sélectionner le programme par année.

Paramètres **year** (*int*) – année (présentement entre 2016 et 2021)

Renvoie Une instance de la classe de l'année sélectionnée.

Type renvoyé class instance

Pour le moment, le programme permet uniquement de calculer les cotisations.

7.2.2 Gabarit du programme

Nous utilisons un gabarit afin de créer les programmes chaque année. Quand le programme change seulement au niveau des paramètres d'une année à l'autre, nous irons chercher seulement les nouveaux paramètres. Quand des fonctions changent, l'utilisateur n'a qu'à modifier les fonctions touchées (ou à en ajouter de nouvelles). L'avantage des classes dérivées est de ne pas avoir à répéter toutes les fonctions d'une année à l'autre si celles-ci n'ont pas changé.

Nous reproduisons ici la spécification du gabarit. Il est basé sur le programme en vigueur en 2016.

class srd.qpip.**template**

Régime québécois d'assurance parentale (RQAP).

Ce gabarit sert pour l'instant à calculer les cotisations au RQAP.

contrib(*p*, *hh*)

Fonction pour calculer les cotisations à l'assurance parentale.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de la cotisation à l'assurance parentale (annuelle).

Type renvoyé float

7.2.3 Fonctions spécifiques ou modifiées par année

class srd.qpip.programs.**program_2016**

Version du programme de 2016.

class srd.qpip.programs.**program_2017**

Version du programme de 2017.

class srd.qpip.programs.**program_2018**

Version du programme de 2018.

class srd.qpip.programs.**program_2019**

Version du programme de 2019.

class srd.qpip.programs.**program_2020**

Version du programme de 2020.

class srd.qpip.programs.**program_2021**

Version du programme de 2021.

CHAPITRE 8

Aide sociale

8.1 Survol

Le programme de l'aide sociale est pris en charge par le module *assistance*. Ce module contient un gabarit que nous documentons ci-dessous ainsi que des classes dérivées spécifiques à chaque année.

La fonction *program* permet de choisir l'année du programme et ira tirer une instance du programme pour cette année. L'instance est retournée par la fonction.

srd.assistance.**program**(*year*)

Fonction qui permet de sélectionner le programme par année.

Paramètres **year** (*int*) – année (présentement entre 2016 et 2020)

Renvoie Une instance de la classe de l'année sélectionnée.

Type renvoyé class instance

Outre la prestation de base, sont pris en compte les suppléments pour personnes seules, la prestation pour contrainte temporaire à l'emploi et les ajustements pour enfants à charge au Québec. Pour l'Ontario, seule la prestation de base est modélisée.

8.2 Gabarit du programme

Nous utilisons un gabarit afin de créer les programmes chaque année. Quand le programme change seulement au niveau des paramètres d'une année à l'autre, nous irons chercher seulement les nouveaux paramètres. Quand des fonctions changent, l'utilisateur n'a qu'à modifier les fonctions touchées (ou à en ajouter de nouvelles). L'avantage des classes dérivées est de ne pas avoir à répéter toutes les fonctions d'une année à l'autre si celles-ci n'ont pas changé.

srd.assistance.**template**()

Classe qui contient un gabarit du programme d'aide sociale (tel qu'il existait en 2016).

À noter que seul un test d'actifs simplifié est appliqué, à un volet ; les actifs liquides (argent comptant et comptes courants) ne sont pas considérés.

Nous présentons ici la liste des fonctions incluses dans le gabarit :
(Cliquez sur le nom pour afficher les détails)

- file()

srd.assistance.program_2016.**file**(*self*, *hh*)

Fonction pour faire une demande au programme et recevoir une prestation.
Cette fonction calcule une prestation intégrée d'aide sociale.

Paramètres **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de l'aide sociale.

Type renvoyé float

- shelter()

srd.assistance.program_2016.**shelter**(*self*, *hh*)

Composante logement.
N'est pas mise en œuvre pour l'instant.

Paramètres **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de la composante logement.

Type renvoyé float

- calc_sa_qc()

srd.assistance.program_2016.**calc_sa_qc**(*self*, *hh*)

Composante de base et supplément pour enfant (en cas de prestation d'ACE réduite) pour le Québec.

Paramètres **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant combiné de la composante de base et du supplément pour enfant.

Type renvoyé float

- eligibility_qc()

srd.assistance.program_2016.**eligibility_qc**(*self*, *hh*)

Fonction qui évalue l'admissibilité de la personne à chacun des 4 volets du programme.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

- child_ajustments()

srd.assistance.program_2016.**child_ajustments**(*self*, *hh*)

Fonction qui calcule l'ajustement des prestations d'aide sociale selon les caractéristiques des enfants du ménage.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- calc_sa_on()

srd.assistance.program_2016.**calc_sa_on**(*self*, *hh*)

Composante de base et supplément pour enfant pour l'Ontario.

Parameters

hh : Hhold instance de la classe Hhold

Renvoie Montant combiné de la composante de base et du supplément pour enfant pour l'Ontario.

Type renvoyé float

8.3 Fonctions spécifiques ou modifiées par année

2016

class srd.assistance.programs.**program_2016**

Version du programme de 2016.

2017

class srd.assistance.programs.**program_2017**

Version du programme de 2017.

2018

class srd.assistance.programs.**program_2018**

Version du programme de 2018.

2019

class srd.assistance.programs.**program_2019**

Version du programme de 2019.

2020

class srd.assistance.programs.**program_2020**

Version du programme de 2020.

2021

class srd.assistance.programs.**program_2021**

Version du programme de 2021.

CHAPITRE 9

Prestations liées à la COVID-19

9.1 Survol

Les mesures d'urgence pour soutenir la population canadienne durant la pandémie de COVID-19 ainsi que les mesures de relance économique sont prises en charge par le module *covid*.

Les mesures prises en compte dans ce module sont :

- La prestation canadienne d'urgence (PCU) ;
- La prestation canadienne d'urgence pour étudiants (PCUE) ;
- La prestation canadienne de la relance économique (PCRE) ;
- Pour le Québec seulement, le programme incitatif pour la rétention des travailleurs essentiels (PIRTE).

9.2 Gabarit du programme

Nous utilisons un gabarit afin de créer les programmes chaque année. Si des fonctions changent, l'utilisateur n'a qu'à modifier les fonctions touchées (ou à en ajouter de nouvelles).

`srd.covid.`**`template`**`()`

Programmes COVID-19.

Ce gabarit sert à calculer les programmes associés à la COVID-19.

Nous présentons ici la liste des fonctions incluses dans le gabarit :
(Cliquez sur le nom pour afficher les détails)

- compute()

`srd.covid.program_2020.`**`compute`**`(`*`self`*`, `*`hh`*`)`

Fonction qui fait le calcul de chaque programme COVID-19.

Paramètres

— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

- compute_cerb()

srd.covid.program_2020.**compute_cerb**(*self*, *p*)

Fonction pour le calcul de la PCU.

Calcule la PCU en fonction du nombre de blocs de 4 semaines (mois) pour lesquels la prestation est demandée.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant de la PCU.

Type renvoyé float

- compute_cesb()

srd.covid.program_2020.**compute_cesb**(*self*, *p*, *hh*)

Fonction pour le calcul de la PCUE.

Calcule la PCUE en fonction de la prestation mensuelle à laquelle l'individu a droit et du nombre de blocs de 4 semaines (mois) pour lesquels la prestation est demandée.

Paramètres
— **p** (Person) – instance de la classe Person
— **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de la PCUE.

Type renvoyé float

- compute_monthly_cesb()

srd.covid.program_2020.**compute_monthly_cesb**(*self*, *p*, *hh*)

Calcule le montant mensuel de la PCUE en fonction du statut (invalidité, dépendants).

Renvoie Prestation mensuelle de PCUE.

Type renvoyé float

- compute_iprew()

srd.covid.program_2020.**compute_iprew**(*self*, *p*)

Fonction pour le calcul du PIRTE.

Calcule la prestation de PIRTE pour la période de 16 semaines (4 mois) si le travailleur est admissible.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant de PIRTE pour les 16 semaines.

Type renvoyé float

- compute_crb()

srd.covid.program_2020.**compute_crb**(*self*, *p*)

Fonction qui calcule la Prestation canadienne de la relance économique (PCRE).

Calcule la PCRE pour la période de 54 semaines si le travailleur est admissible.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant de la PCRE.

Type renvoyé float

9.3 Fonctions spécifiques ou modifiées par année

2020

class srd.covid.programs.**program_2020**

Version du programme de 2020.
Calcul des prestations liées à la COVID-19 : la Prestation canadienne d'urgence (PCU), la Prestation canadienne d'urgence pour les étudiants (PCUE), le Programme incitatif pour la rétention des travailleurs essentiels (PIRTE) au Québec et la Prestation canadienne de la relance économique (PCRE).

- compute_cerb()

srd.covid.programs.program_2020.**compute_cerb**(*self*, *p*)

Fonction pour le calcul de la PCU.
Calcule la PCU en fonction du nombre de blocs de 4 semaines (mois) pour lesquels la prestation est demandée.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant de la PCU.

Type renvoyé float

- compute_cesb()

srd.covid.programs.program_2020.**compute_cesb**(*self*, *p*, *hh*)

Fonction pour le calcul de la PCUE.
Calcule la PCUE en fonction de la prestation mensuelle à laquelle l'individu a droit et du nombre de blocs de 4 semaines (mois) pour lesquels la prestation est demandée.

Paramètres
- **p** (Person) – instance de la classe Person
- **hh** (Hhold) – instance de la classe Hhold

Renvoie Montant de la PCUE.

Type renvoyé float

- compute_monthly_cesb()

srd.covid.programs.program_2020.**compute_monthly_cesb**(*self*, *p*, *hh*)

Calcule le montant mensuel de la PCUE en fonction du statut (invalidité, dépendants).

Renvoie Prestation mensuelle de PCUE.

Type renvoyé float

- compute_iprew()

srd.covid.programs.program_2020.**compute_iprew**(*self*, *p*)

Fonction pour le calcul du PIRTE.
Calcule la prestation de PIRTE pour la période de 16 semaines (4 mois) si le travailleur est admissible.

Paramètres **p** (Person) – instance de la classe Person

Renvoie Montant de PIRTE pour les 16 semaines.

Type renvoyé float

2021

class srd.covid.programs.**program_2021**

Version du programme de 2021.
Calcule principalement la Prestation canadienne de la relance économique (PCRE).

CHAPITRE 10

Calculateur de revenu disponible

10.1 Survol

La classe principale du simulateur est *tax* qui permet de faire différents calculs d'impôts, de prestations et de cotisations et de calculer le revenu disponible ainsi que des taux moyens et des taux marginaux effectifs d'imposition.

10.2 Les fonctions du calculateur

Voici les différentes fonctions de calcul incluses dans la classe principale.

class srd.**tax**(*year*, *ifed=True*, *ioas=True*, *iprov=True*, *ipayroll=True*, *iass=True*)

Classe générale pour le calcul des impôts, cotisations et prestations.

Paramètres

- **year** (*int*) – année pour le calcul
- **ifed** (*boolean*) – vrai si le calcul de l'impôt fédéral est demandé
- **ioas** (*boolean*) – vrai si le calcul des prestations de PSV, SRG, Allocation et Allocation au survivant est demandé
- **iprov** (*boolean*) – vrai si le calcul de l'impôt provincial est demandé
- **ipayroll** (*boolean*) – vrai si le calcul des cotisations sociales est demandé
- **iass** (*boolean*) – vrai si le calcul des prestations d'aide sociale est demandé

- compute()

srd.tax.**compute**(*self*, *hh*, *n_points=1*)

Cette fonction transfère des revenus de pension pour les couples admissibles et retient la solution qui maximise le revenu disponible familial. Si n_points=0, pas de fractionnement des revenus de pension. Par défaut (n_points=1), les revenus bruts sont égalisés dans la mesure des transferts possibles. Pour n>1, une simulation est faite pour chaque point de la grille. À noter que lorsque n augmente, les solutions avec n inférieur (notamment n=0) sont aussi considérées.

Paramètres

- **hh** (Hhold) – instance de la classe Hhold

— **n_points** (*int*) – nombre de points utilisés pour optimiser le fractionnement de revenus de pension

- compute_with_transfer()

srd.tax.**compute_with_transfer**(*self*, *hh*, *transfer*)

Cette fonction effectue les transferts de revenus de pension et appelle la fonction qui simule le ménage.

Paramètres
— **hh** (*Hold*) – instance de la classe Hhold
— **transfer** (*float*) – transfert du premier au second conjoint (du second au premier si négatif)

- compute_all()

srd.tax.**compute_all**(*self*, *hh*)

Calcule tous les éléments demandés.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- compute_oas()

srd.tax.**compute_oas**(*self*, *hh*)

Calcul des prestations de PSV, SRG, Allocation et Allocation au survivant.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- compute_fed()

srd.tax.**compute_fed**(*self*, *hh*)

Calcul de l'impôt fédéral.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- compute_prov()

srd.tax.**compute_prov**(*self*, *hh*)

Calcul de l'impôt provincial.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- compute_payroll()

srd.tax.**compute_payroll**(*self*, *hh*)

Calcul des cotisations sociales.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- compute_covid()

srd.tax.**compute_covid**(*self*, *hh*)

Calcul de la PCU, de la PCRE, de la PCUE et du PIRTE.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- compute_sa()

srd.tax.**compute_sa**(*self*, *hh*)

Calcul des prestations d'aide sociale.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- compute_after_tax_inc()

srd.tax.**compute_after_tax_inc**(*self*, *hh*)

Calcul du revenu après impôt fédéral et provincial.

Calcul fait au niveau individuel et ensuite rattaché à la personne ; le résultat au niveau du ménage est aussi disponible.

Paramètres **hh** (Hhold) – instance de la classe Hhold

- disp_inc()

srd.tax.**disp_inc**(*self*, *hh*)

Calcul du revenu disponible après impôts, cotisations sociales, épargne (positive ou négative) et prestations.

Calcul fait au niveau individuel et ensuite rattaché à la personne ; le résultat au niveau du ménage est aussi disponible.

Paramètres **hh** (Hhold) – instance de la classe Hhold

CHAPITRE 11

Comparaisons avec le MFQ

Cette section compare brièvement les résultats du SRD avec ceux du simulateur de revenu disponible du ministère des Finances du Québec (MFQ) . L'objectif poursuivi est de démontrer que le SRD est en mesure de produire des résultats comparables à ceux du MFQ.

Deux indicateurs sont utilisés pour cette comparaison. Le premier est le revenu disponible, qui est défini comme le revenu total (incluant les transferts directs gouvernementaux) après soustraction des cotisations salariales et de l'impôt sur le revenu, et ajout des transferts sociaux. Le deuxième indicateur est le taux effectif marginal d'imposition (TEMI), qui est une mesure du taux de charge fiscale sur un revenu supplémentaire, soit le taux qui s'applique à la dernière tranche de revenu imposable obtenu par un particulier (nous procedons par tranche de 1 000 $).

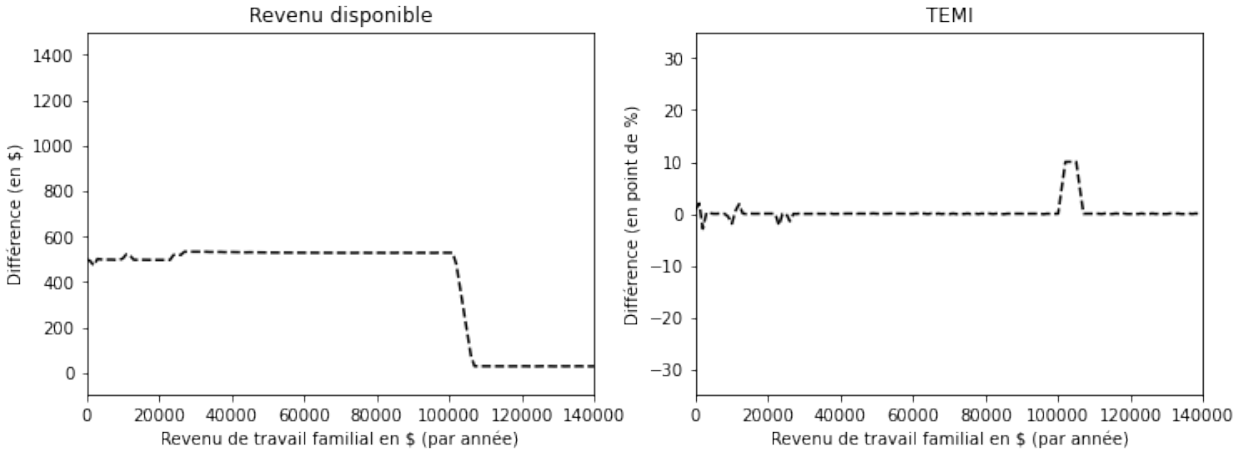
De manière générale, les résultats obtenus par le SRD sont très similaires à ceux du MFQ. Notons que lors de la modélisation, nous avons choisi d'intégrer le crédit d'impôt attribuant un montant ponctuel (de 500$) pour pallier à la hausse du coût de la vie en 2021, à la différence du MFQ qui l'a intégré en 2022. Ce choix donne lieu à une différence de 500$ à 1000$ dépendant de la situation du ménage (célibataire ou couple). En outre, d'autres légères différences sont observables. Celles-ci sont principalement dues aux éléments suivants :

- les résultats du MFQ tiennent compte du changement d'année de base en juillet pour le calcul du crédit d'impôt de la TPS, alors que le SRD ne considère qu'une seule année de base pour une même année fiscale ;
- l'aide sociale n'est pas incluse dans le revenu familial lors du calcul de la prime au travail, puisque le montant de l'aide sociale est calculé après la prime au travail ;
- on observe une légère différence (positive) pour la contribution au régime public d'assurance médicaments pour les 65 ans et plus.

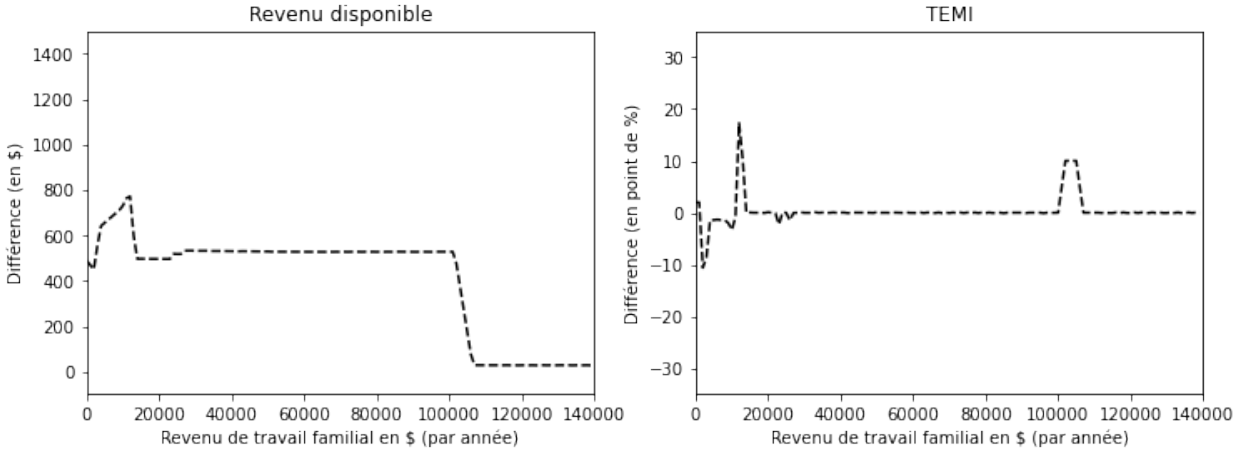
Dix cas-types sont présentés ci-dessous afin de dresser un portrait général de la situation. Les différences correspondent aux résultats du SRD moins ceux du MFQ. L'année fiscale 2021 a été choisie aux fins de comparaison et l'allocation logement est exclue du revenu disponible, car celle-ci n'est pas modélisée par le SRD.

11.1 Personne célibataire (sans enfant)

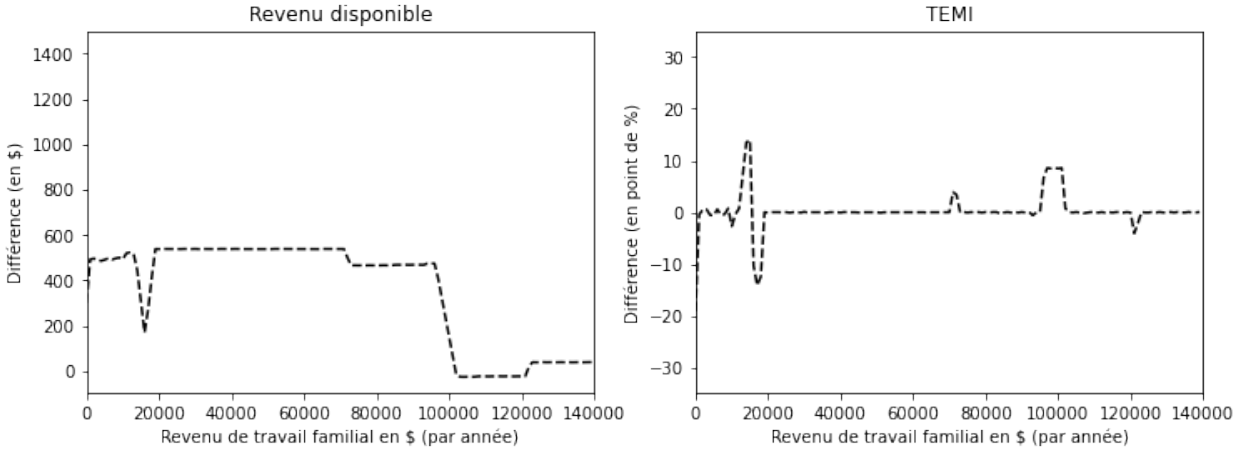
De 35 ans



De 63 ans

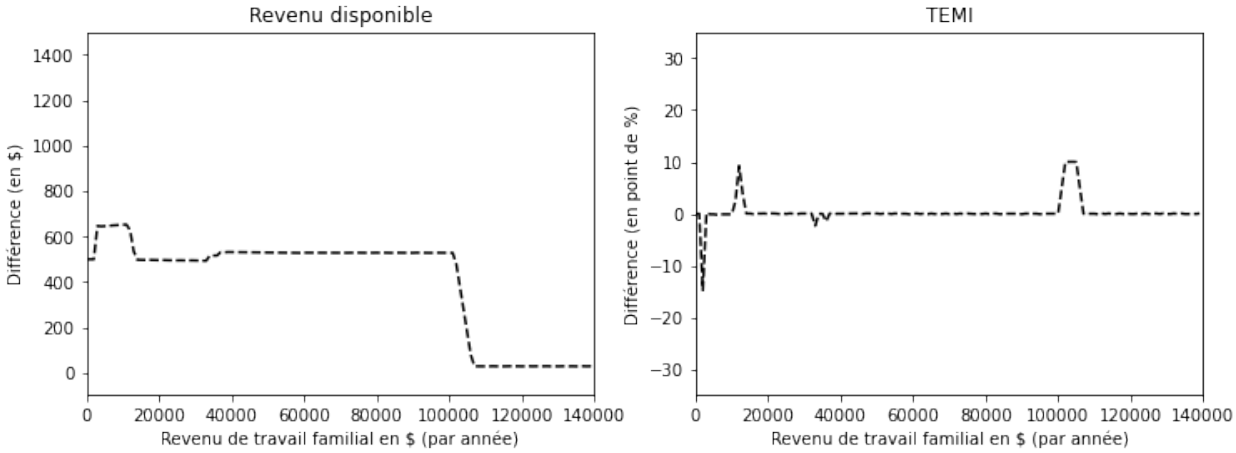


De 73 ans

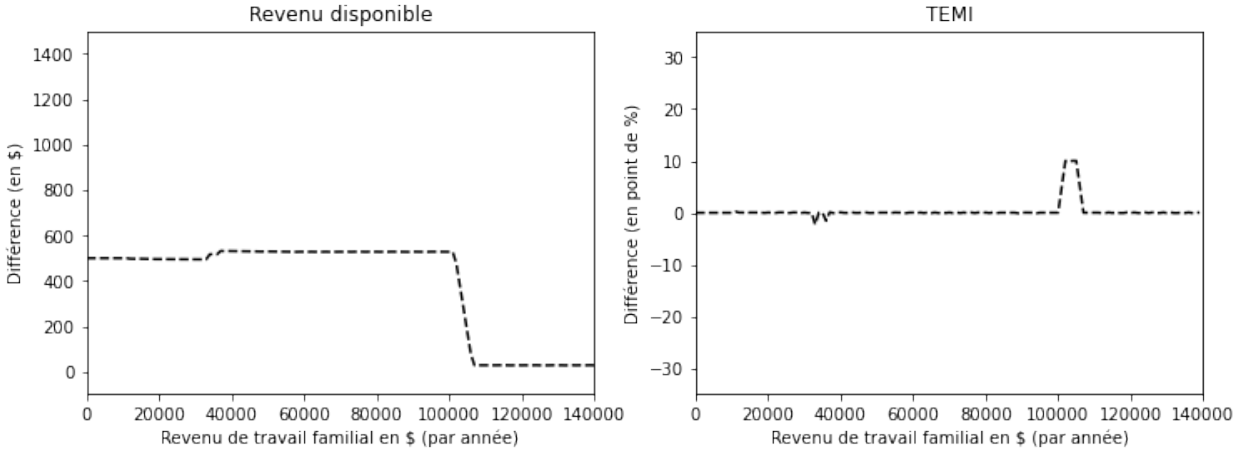


11.2 Personne monoparentale

De 35 ans avec un enfant de 4 ans et des frais de garde non-subventionnés de 6825$

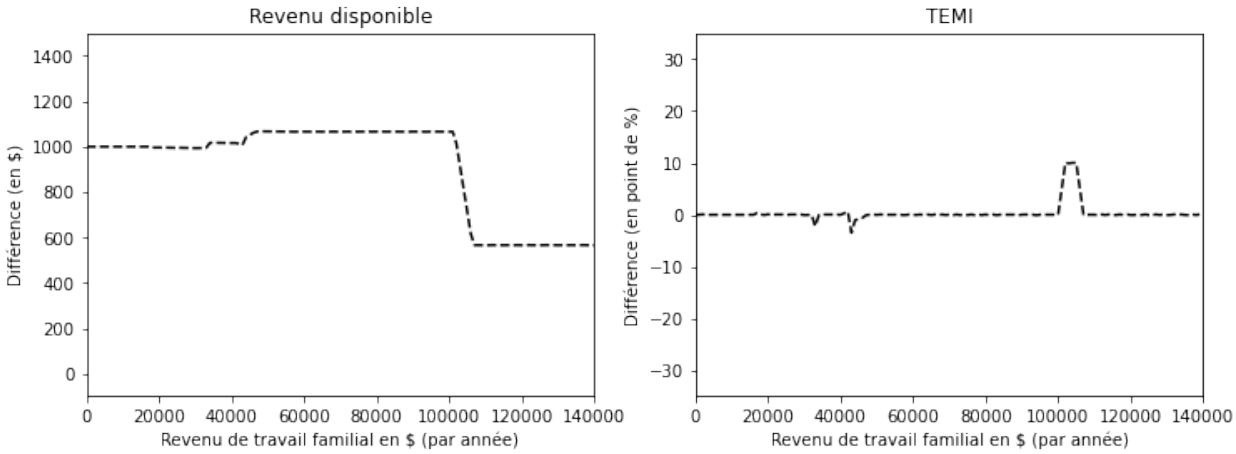


De 35 ans avec un enfant de 10 ans, aucun frais de garde

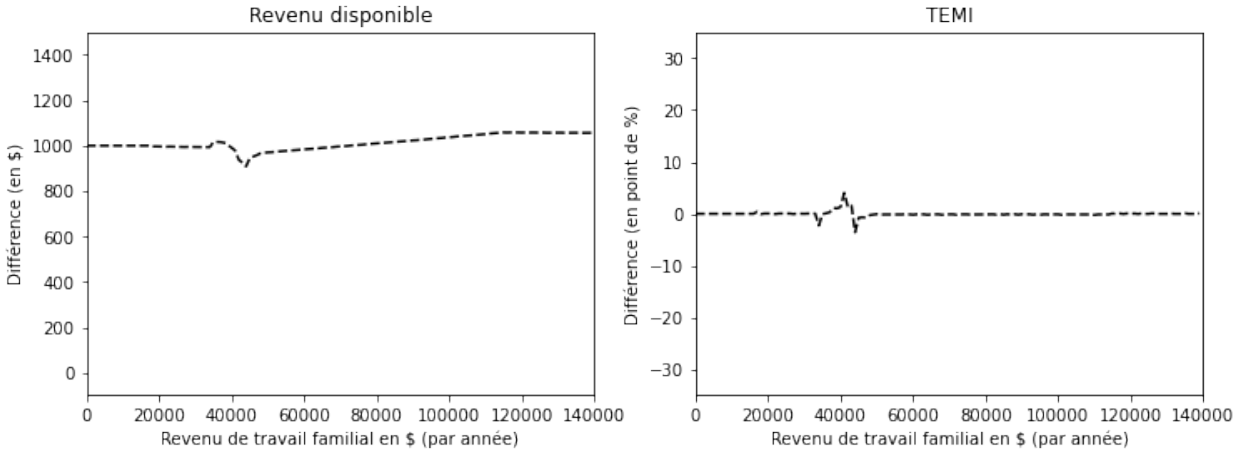


11.3 Couple sans enfant

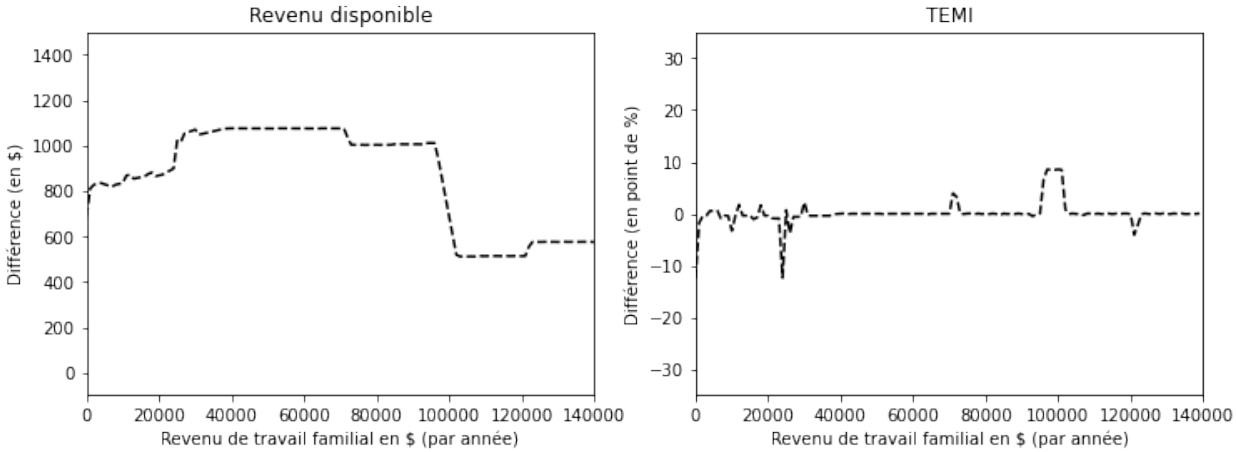
Conjoints de 35 ans et répartition du revenu familial de 0-100% entre les conjoints



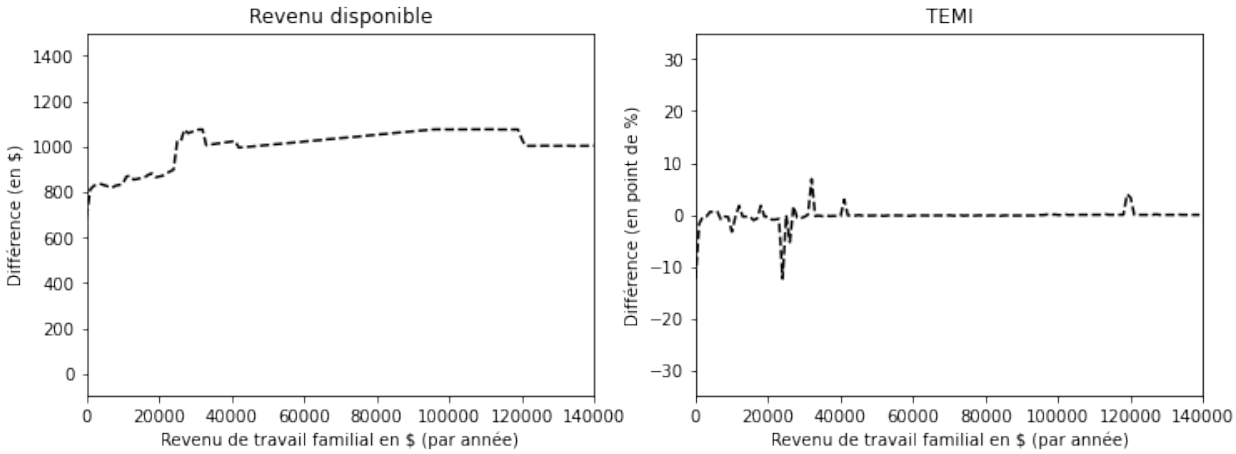
Conjoints de 35 ans et répartition du revenu familial de 40-60% entre les conjoints



Conjoints de 73 ans et répartition du revenu 0-100% entre les conjoints

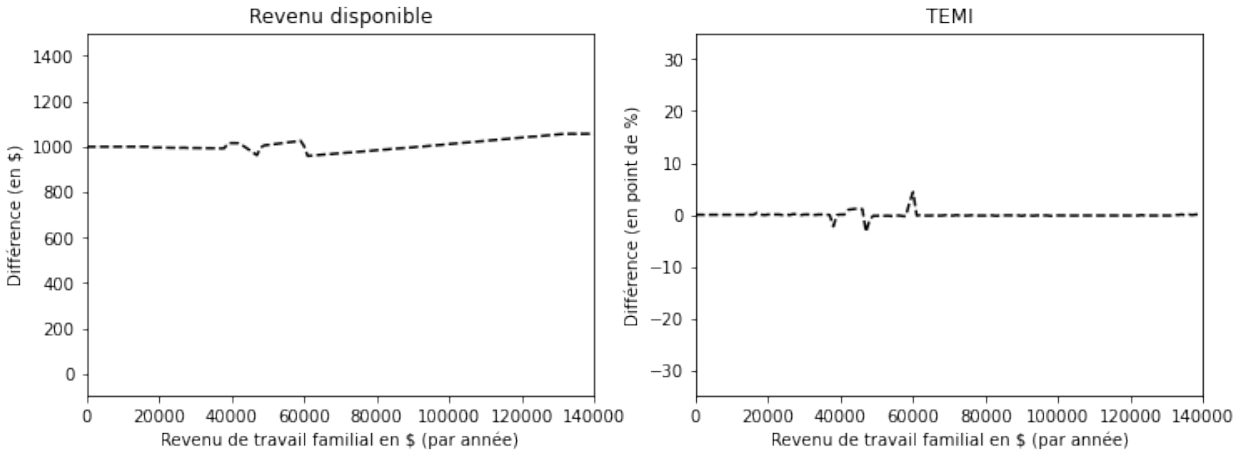


Conjoints de 73 ans et répartition du revenu familial de 40-60% entre les conjoints



11.4 Couple avec enfant

Conjoints de 35 ans avec un enfant de 4 ans et des frais de garde non-subventionnés de 6825$ et répartition du revenu familial de 40-60% entre les conjoints



CHAPITRE 12

Exemple de calcul de l'impôt fédéral

12.1 Importation du package

```
import srd
```

12.2 Initialisation d'un ménage

On doit d'abord initialiser un ménage. Ici nous supposerons un couple avec deux membres ayant tous les deux 45 ans et des revenus de travail de 50 000$ et 25 000$, respectivement.

```
jean = srd.Person(age=45,earn=50e3)
pauline = srd.Person(age=45,earn=25e3)
```

On les insère dans un ménage vivant au Québec.

```
hh = srd.Hhold(jean,pauline,prov='qc')
```

On peut voir le profil de chacun des membres du ménage en utilisant *vars()* :

```
vars(jean)
```

```
'inc_earn': 50000.0,
'inc_self_earn': 0,
'inc_work_month': [4166.666666666667,
4166.666666666667,
4166.666666666667,
4166.666666666667,
4166.666666666667,
4166.666666666667,
```

(suite sur la page suivante)

(suite de la page précédente)

```
4166.666666666667,
4166.666666666667,
4166.666666666667,
4166.666666666667,
4166.666666666667,
4166.666666666667],
'prev_inc_work': 50000.0,
'inc_rpp': 0,
'inc_cpp': 0,
'net_cap_gains': 0,
'prev_cap_losses': 0,
'cap_gains_exempt': 0,
'inc_othtax': 0,
'inc_othntax': 0,
'div_elig': 0,
'div_other_can': 0,
'inc_rrsp': 0,
'con_rrsp': 0,
'con_non_rrsp': 0,
'con_rpp': 0,
'union_dues': 0,
'donation': 0,
'gift': 0,
'years_can': 45,
'disabled': False,
'widow': False,
'med_exp': 0,
'ndays_chcare_k1': 0,
'ndays_chcare_k2': 0,
'asset': 0,
'months_crb': 0,
'months_ei': 0,
'oas_years_post': 0,
'months_cesb': 0,
'months_cerb': 0,
'pub_drug_insurance': False,
'student': False,
'essential_worker': False,
'emp_temp_constraints': False,
'hours_month': None,
'dep_senior': False,
'home_support_cost': 0,
'pension_split': 0,
'pension_split_qc': 0,
'pension_deduction': 0,
'pension_deduction_qc': 0,
'inc_oas': 0,
'inc_gis': 0,
'inc_ei': 0,
'inc_sa': 0,
'allow_couple': 0,
'allow_surv': 0,
```

(suite sur la page suivante)

(suite de la page précédente)

```
'inc_cerb': 0,
'inc_cesb': 0,
'inc_crb': 0,
'inc_iprew': 0,
'covid': None,
'after_tax_inc': None,
'disp_inc': None,
'fed_return': None,
'prov_return': None,
'payroll': None,
'max_split': 0.0}
```

12.3 Calcul de l'impôt fédéral

On doit d'abord créer un formulaire d'impôt pour une année en particulier.

```
from srd import federal
fed_form = federal.form(2021)
```

On peut voir les différents paramètres du système fiscal en utilisant encore *vars()* :

```
vars(fed_form)
```

```
{'div_elig_factor': 1.38,
'div_other_can_factor': 1.15,
'div_elig_cred_rate': 0.150198,
'div_other_can_cred_rate': 0.090313,
'cap_gains_rate': 0.5,
'qpip_deduc_rate': 0.43786,
'basic_amount_poor': 13808.0,
'basic_amount_rich': 12421.0,
'age_cred_amount': 7713.0,
'min_age_cred': 65,
'age_cred_exempt': 38893.0,
'age_cred_claw_rate': 0.15,
'empl_cred_max': 1257.0,
'pension_cred_amount': 2000.0,
'pension_cred_min_age_split': 65,
'disability_cred_amount': 8662.0,
'med_exp_nr_cred_max_age': 18,
'med_exp_nr_cred_max_claw': 2421.0,
'med_exp_nr_cred_rate': 0.03,
'donation_frac_net': 0.75,
'donation_low_cut': 200.0,
'donation_high_cut': 216511.0,
'donation_low_rate': 0.15,
'donation_med_rate': 0.29,
'donation_high_rate': 0.33,
'rate_non_ref_tax_cred': 0.15,
'rate_abatment_qc': 0.165,
```

(suite sur la page suivante)

(suite de la page précédente)

```
'ccb_young': 6833.0,
'ccb_old': 5765.0,
'ccb_covid_supp': 300.0,
'ccb_max_num_ch': 4,
'ccb_cutoff_1': 32028.0,
'ccb_cutoff_2': 69395.0,
'ccb_rate_1_1ch': 0.07,
'ccb_rate_1_2ch': 0.135,
'ccb_rate_1_3ch': 0.19,
'ccb_rate_1_4ch': 0.23,
'ccb_rate_2_1ch': 0.032,
'ccb_rate_2_2ch': 0.057,
'ccb_rate_2_3ch': 0.08,
'ccb_rate_2_4ch': 0.095,
'ccb_ccbycs_cutoff': 120000.0,
'ccb_ccbycs_1': 1200.0,
'ccb_ccbycs_2': 600.0,
'med_exp_rate': 0.25,
'med_exp_claw_rate': 0.05,
'med_exp_claw_cutoff': 28446.0,
'med_exp_max': 1285.0,
'med_exp_min_work_inc': 3715.0,
'gst_cred_claw_rate': 0.05,
'gst_cred_claw_cutoff': 38892.0,
'gst_cred_base': 299.0,
'gst_cred_other': 157.0,
'gst_cred_rate': 0.02,
'gst_cred_base_amount': 9686.0,
'gst_covid_single': 400.0,
'gst_covid_couple': 600.0,
'chcare_young': 8000.0,
'chcare_old': 5000.0,
'chcare_rate_inc': 0.66666,
'ei_max_net_inc': 70375.0,
'ei_rate_repay': 0.3,
'ded_qpp_rate': 0.052632,
'ded_qpp_exempt': 3500,
'ded_qpp_claw_rate': 0.03,
'nrtc_spouse_dis': 2296,
'dep_disa_amount': 2296,
'base_single': 2400,
'base_couple': 3600,
'rate_single_dep': 0.15,
'rate': 0.274,
'rate_couple_dep': 0.14,
'max_single': 2348.36,
'max_single_dep': 1285.6,
'max_couple': 3665.7,
'max_couple_dep': 1872.99,
'exempt_single': 12463.5,
'exempt_single_dep': 12459.84,
'exempt_couple': 19135.61,
```

(suite sur la page suivante)

(suite de la page précédente)

```
'exempt_couple_dep': 18143.56,
'claw_rate': 0.2,
'dis_base': 1200,
'dis_max': 721.01,
'dis_rate_single': 0.4,
'dis_rate_couple': 0.2,
'dis_exempt_single': 24205.3,
'dis_exempt_single_dep': 18887.84,
'dis_exempt_couple': 37464.11,
'dis_exempt_couple_dep': 28508.51,
'dis_claw_rate': 0.2,
'dis_couple_claw_rate': 0.1,
'l_brackets': [0.0, 49020.0, 98040.0, 151978.0, 216511.0],
'l_rates': [0.15, 0.205, 0.26, 0.29, 0.33],
'l_constant': [0.0, 7353.0, 17402.0, 31425.0, 50140.0],
'witb_params': {'on': {'base_single': 3000,
'base_couple': 3000,
'rate_single_dep': 0.27,
'rate': 0.27,
'rate_couple_dep': 0.27,
'max_single': 1395.0,
'max_single_dep': 2403.0,
'max_couple': 2403.0,
'max_couple_dep': 2403.0,
'exempt_single': 22944.0,
'exempt_single_dep': 26177.0,
'exempt_couple': 26177.0,
'exempt_couple_dep': 26177.0,
'claw_rate': 0.12,
'dis_base': 1150,
'dis_max': 720.0,
'dis_rate_single': 0.27,
'dis_rate_couple': 0.27,
'dis_exempt_single': 32244.0,
'dis_exempt_single_dep': 42197.0,
'dis_exempt_couple': 42197.0,
'dis_exempt_couple_dep': 42197.0,
'dis_claw_rate': 0.15,
'dis_couple_claw_rate': 0.075},
'qc': {'base_single': 2400,
'base_couple': 3600,
'rate_single_dep': 0.15,
'rate': 0.274,
'rate_couple_dep': 0.14,
'max_single': 2348.36,
'max_single_dep': 1285.6,
'max_couple': 3665.7,
'max_couple_dep': 1872.99,
'exempt_single': 12463.5,
'exempt_single_dep': 12459.84,
'exempt_couple': 19135.61,
'exempt_couple_dep': 18143.56,
```

(suite sur la page suivante)

(suite de la page précédente)

```
'claw_rate': 0.2,
'dis_base': 1200,
'dis_max': 721.01,
'dis_rate_single': 0.4,
'dis_rate_couple': 0.2,
'dis_exempt_single': 24205.3,
'dis_exempt_single_dep': 18887.84,
'dis_exempt_couple': 37464.11,
'dis_exempt_couple_dep': 28508.51,
'dis_claw_rate': 0.2,
'dis_couple_claw_rate': 0.1}}}
```

On remplit le formulaire d'impôt à l'aide de la fonction *tax()*.

```
from srd import tax
tax_form = tax(2021)
tax_form.compute(hh)
```

On peut visualiser un formulaire d'impôt sommaire qui est rattaché à chaque personne :

```
jean.fed_return
```

```
{'gross_income': 50000.0,
'deductions_gross_inc': 232.5,
'net_income': 49767.5,
'deductions_net_inc': 0.0,
'taxable_income': 49767.5,
'gross_tax_liability': 7506.2375,
'non_refund_credits': 2763.45,
'refund_credits': 782.5599375,
'net_tax_liability': 3960.2275625}
```

```
pauline.fed_return
```

```
{'gross_income': 25000.0,
'deductions_gross_inc': 107.5,
'net_income': 24892.5,
'deductions_net_inc': 0.0,
'taxable_income': 24892.5,
'gross_tax_liability': 3733.875,
'non_refund_credits': 2497.4249999999997,
'refund_credits': 204.01425000000006,
'net_tax_liability': 1032.43575}
```

On peut ajouter des enfants au ménage à l'aide de la fonction *add_dependent()*.

```
emma = srd.Dependent(age=4, child_care=2000, med_exp=500)
alex = srd.Dependent(age=14, school=4000)
hh.add_dependent(emma,alex)
```

On peut ensuite calculer le nouveau formulaire d'impôt pour les deux adultes de la famille.

```
tax_form.compute(hh)
jean.fed_return
```

```
{'gross_income': 50000.0,
'deductions_gross_inc': 232.5,
'net_income': 49767.5,
'deductions_net_inc': 0.0,
'taxable_income': 49767.5,
'gross_tax_liability': 7506.2375,
'non_refund_credits': 2763.45,
'refund_credits': 5066.2349375,
'net_tax_liability': -323.4474375}
```

```
pauline.fed_return
```

```
{'gross_income': 25000.0,
'deductions_gross_inc': 2107.5,
'net_income': 22892.5,
'deductions_net_inc': 0.0,
'taxable_income': 22892.5,
'gross_tax_liability': 3433.875,
'non_refund_credits': 2497.4249999999997,
'refund_credits': 4438.18925,
'net_tax_liability': -3501.73925}
```

On peut voir les différents attributs du ménage en utilisant encore *vars()* :

```
vars(hh)
```

```
{'sp': [<srd.actors.Person at 0x11d555880>,
<srd.actors.Person at 0x11d555b20>],
'couple': True,
'prov': 'qc',
'dep': [<srd.actors.Dependent at 0x11d71daf0>,
<srd.actors.Dependent at 0x11d71da90>],
'nkids_0_6': 1,
'nkids_7_16': 1,
'nkids_0_17': 2,
'nkids_0_18': 2,
'n_adults_in_hh': 2,
'elig_split': False,
'sa_elig_asset': True,
'nkids_0_5': 1,
'nkids_6_17': 1}
```

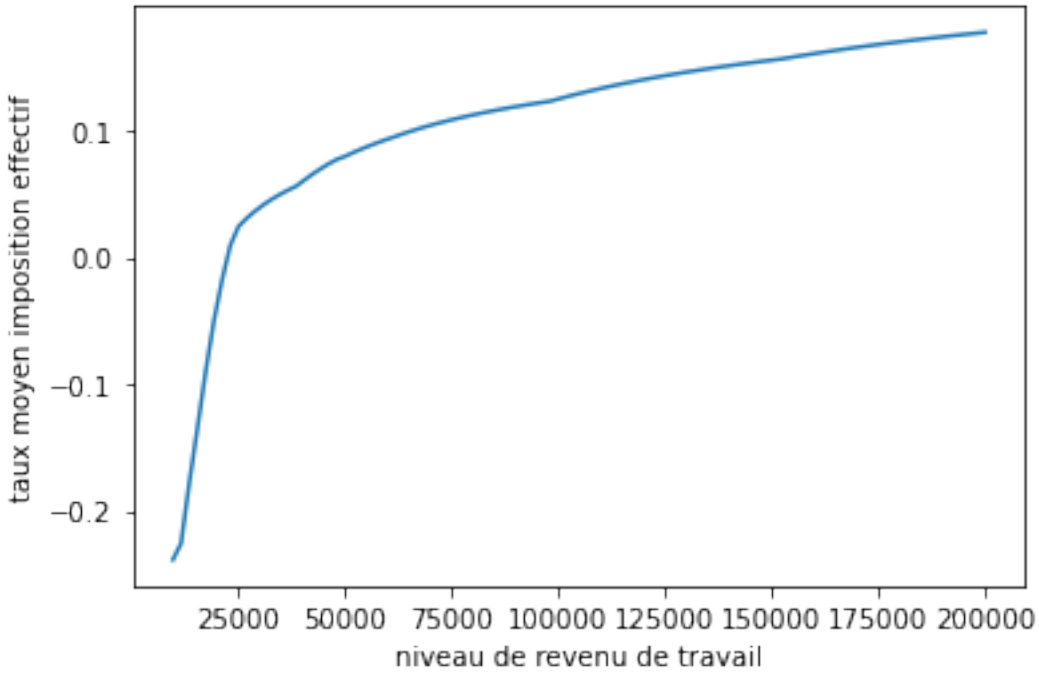
12.4 Expériences

On peut faire des expériences assez complexes. Une première consisterait par exemple à regarder l'impôt fédéral payé si on augmente les revenus de travail par petites tranches.

```
import numpy as np
from matplotlib import pyplot as plt

earns = np.linspace(10e3,200e3,100)
atrs = []
for earn in earns:
    jean = srd.Person(age=45,earn=earn)
    hh = srd.Hhold(jean, prov='qc')
    tax_form.compute(hh)
    atrs.append(jean.fed_return['net_tax_liability']/jean.fed_return['gross_income'])

plt.figure()
plt.plot(earns,atrs)
plt.xlabel('niveau de revenu de travail')
plt.ylabel('taux moyen imposition effectif')
plt.show()
```



On peut aussi faire une expérience dans laquelle on change un paramètre du système d'imposition. Supposons par exemple qu'on augmente le montant de base :

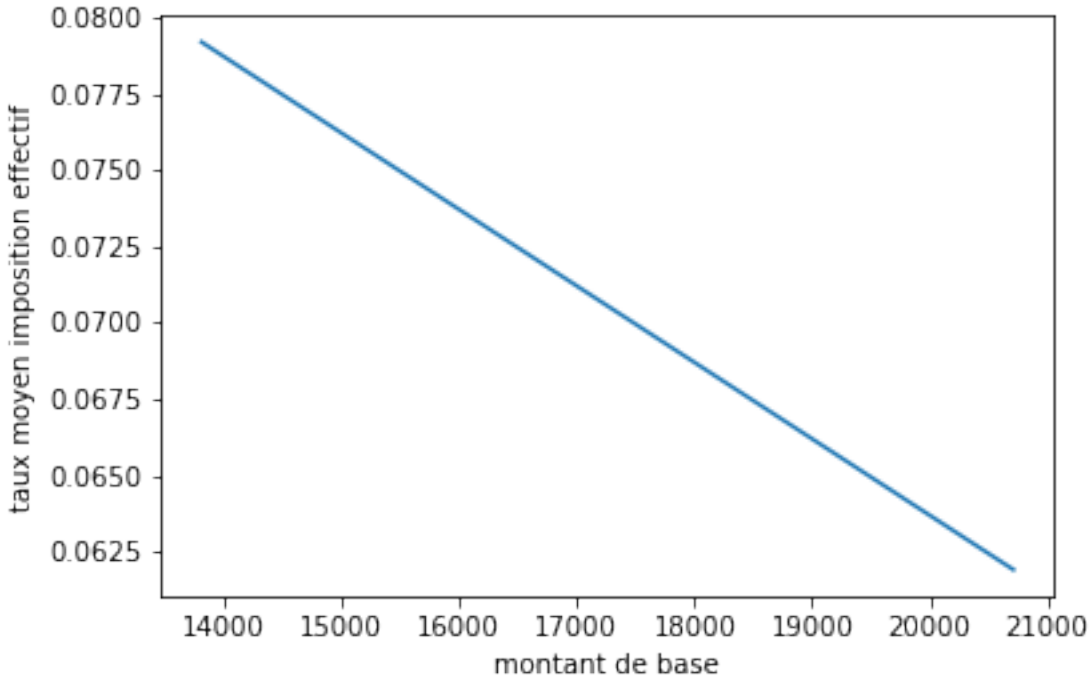
```
base = np.linspace(1.0,1.5,10)
atrs = []
bases = []
jean.inc_earn = 50e3
base_amount = tax_form.federal.basic_amount_poor
```

(suite sur la page suivante)

(suite de la page précédente)

```
for b in base:
    tax_form.federal.basic_amount_poor = base_amount * b
    bases.append(tax_form.federal.basic_amount_poor)
    tax_form.compute(hh)
    atrs.append(jean.fed_return['net_tax_liability']/jean.fed_return['gross_income'])

plt.figure()
plt.plot(bases,atrs)
plt.xlabel('montant de base')
plt.ylabel('taux moyen imposition effectif')
plt.show()
```



CHAPITRE 13

Exemple de calcul du revenu disponible

13.1 Construction du ménage

On construit un ménage où les deux conjoints ont 45 ans, les revenus de travail sont respectivement 50 000$ et 25 000$ et Jean contribue 5000$ à son REER.

```
import srd
```

```
jean = srd.Person(age=45,earn=50e3,con_rrsp=5e3)
pauline = srd.Person(age=45,earn=25e3)
hh = srd.Hhold(jean,pauline,prov='qc')
```

Le revenu familial total, disponible et après impôts peuvent être appelés. À ce stade, le revenu après impôts et le revenu disponible ne sont pas encore calculés.

```
hh.fam_inc_tot, hh.fam_after_tax_inc, hh.fam_disp_inc
```

```
(75000.0, None, None)
```

13.2 Le calculateur

On invoque une instance du calculateur en spécifiant l'année. Il existe aussi des indicateurs pour spécifier si on veut ou non obtenir le calcul de différents impôts. L'exemple ici utilisera seulement le calcul de l'impôt fédéral.

```
tax_form = srd.tax(2021)
```

La fonction *compute()* calcule tous les impôts et cotisations demandés :

```
tax_form.compute(hh)
```

13.3 Calcul du revenu après impôts et du revenu disponible

Après avoir fait les calculs, on peut aussi calculer différents concepts de revenu après impôts à l'aide des fonctions *compute_after_tax_inc()* et *disp_inc()*. Celles-ci viendront modifier les attributs des conjoints pour les variables *after_tax_inc* et *disp_inc*.

```
tax_form.disp_inc(hh)
```

```
tax_form.compute_after_tax_inc(hh)
```

On peut voir que les variables ont été modifiées :

```
hh.fam_inc_tot, hh.fam_after_tax_inc, hh.fam_disp_inc
```

```
(75000.0, 65298.815624999996, 55016.315624999996)
```

```
jean.inc_tot, jean.after_tax_inc, jean.disp_inc
```

```
(50000.0, 42525.176374999995, 33934.676374999995)
```

```
pauline.inc_tot, pauline.after_tax_inc, pauline.disp_inc
```

```
(25000.0, 22773.63925, 21081.63925)
```

CHAPITRE 14

Exemple de changement de paramètre

Cliquez ici pour accéder au notebook.

CHAPITRE 15

Contacts et droits d'utilisation

15.1 Liste d'envoi

Pour rester informé.e des mises à jour du SRD (environ 1 à 2 fois par an), inscrivez-vous à notre liste d'envoi dédiée.

15.2 Personne-contact

Nicholas-James Clavet

15.3 Contributeurs

15.3.1 Principaux

David Boisclair, Nicholas-James Clavet, Roger Edindali Emone, Raquel Fonseca, Pierre-Carl Michaud, Pierre-Yves Yanni

15.3.2 2020

Marianne Laurin, Paul Ouimet, Gaëlle Simard-Duplain

15.4 Droits d'utilisation

Le SRD est fourni sous licence MIT (« MIT License »). Les conditions de la licence sont les suivantes :

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the « Software »), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions :

The copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED « AS IS », WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

CHAPITRE 16

Index

— genindex

CHAPITRE 17

Documentation SRD en PDF

Documentation `pdf`

Index des modules Python

S

Index

D

E

F

G

H

I

L

M

O

P

Q

R

S

T

W