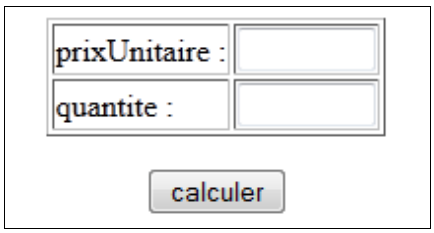
PluriAlgo pour un utilisateur de Javascript

Avant propos

En schématisant à l'extrême, PluriAlgo gère deux types de langages :

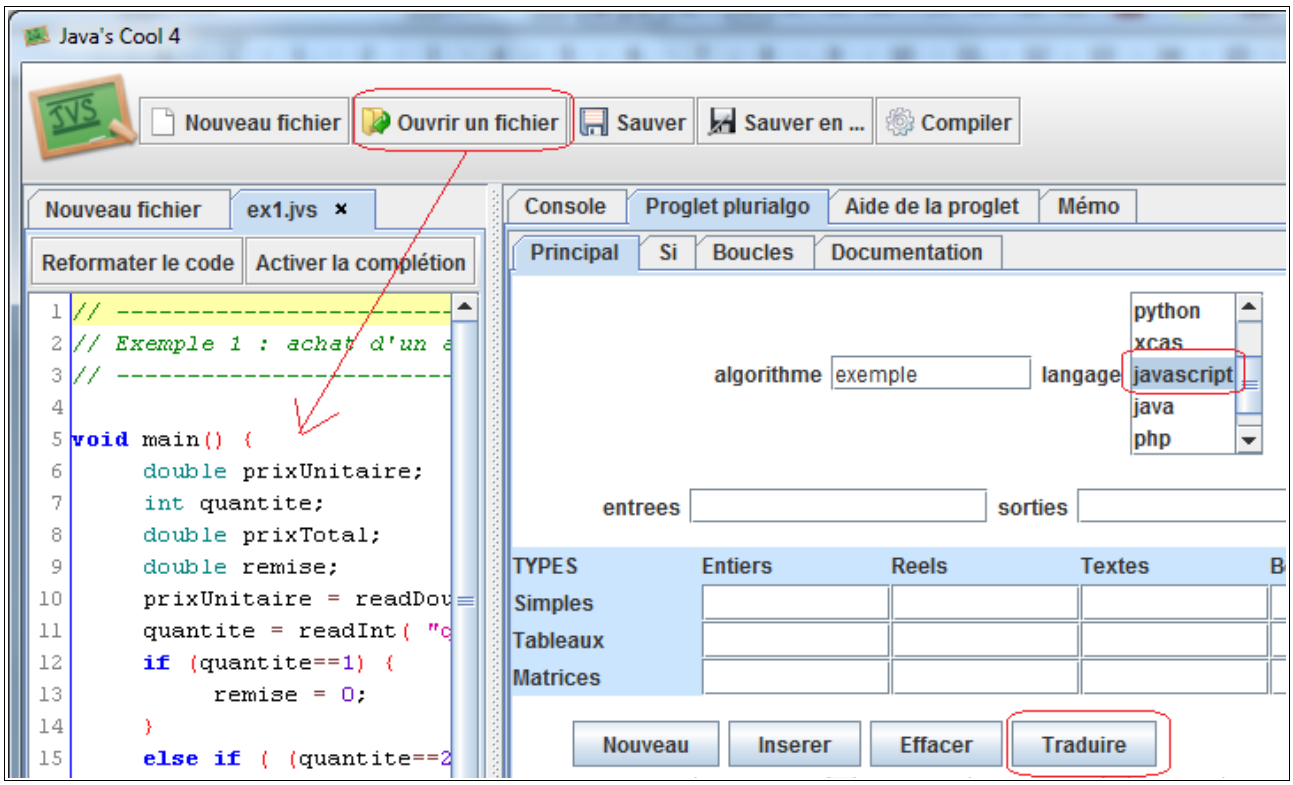
- des langages pédagogiques (Javascool, Algobox, Python...), adaptés à une initiation à l'algorithmique, pour lesquels de nombreux outils sont proposés.
- des langages plus techniques (Java, Php...), généralement abordés après l'étude d'au moins un langage pédagogique, pour lesquels moins d'outils sont disponibles.

Même si Javascript est parfois utilisé comme langage d'initiation, j'ai décidé de privilégier son côté technique dans l'implémentation de PluriAlgo. Par exemple, PluriAlgo regroupe (si possible) les saisies dans un formulaire html :



Le code étant plus complexe qu'avec des saisies séparées (fonction « prompt » de Javascript), il devient difficile (voire impossible pour l'exemple 2) de réutiliser certains outils illustrés dans l'article principal. C'est pourquoi une autre approche est mise en œuvre : appliquer le traducteur de PluriAlgo aux solutions Javascool.

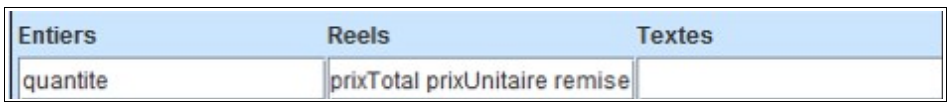
Exemple 1 : achat d'un article



La solution Javascool à traduire est disponible dans le fichier zippé, joint à l'article, contenant tous les programmes développés. Après l'avoir chargée dans l'éditeur, il faut fixer le langage de traduction (ici Javascript) dans l'onglet Principal et cliquer sur le bouton **Traduire**.

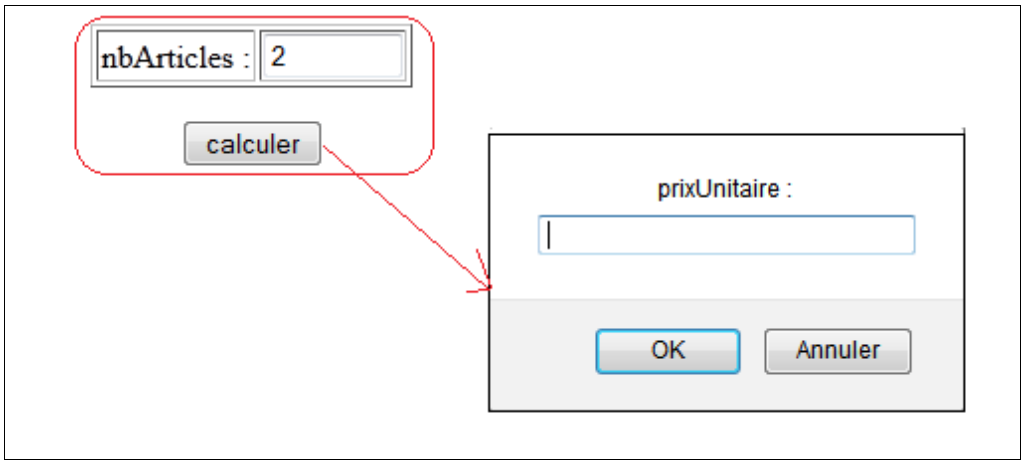
> Remarque : le résultat obtenu est ici opérationnel, sans qu'aucune adaptation ne soit nécessaire, parce que le programme initial est simple et ne comporte aucune fonction prédéfinie.

Il est également possible d'appliquer le traducteur aux solutions Algobox ou Python. Avec Python, langage où les variables sont déclarées implicitement, il est nécessaire de compléter l'onglet Principal avant d'effectuer la traduction :

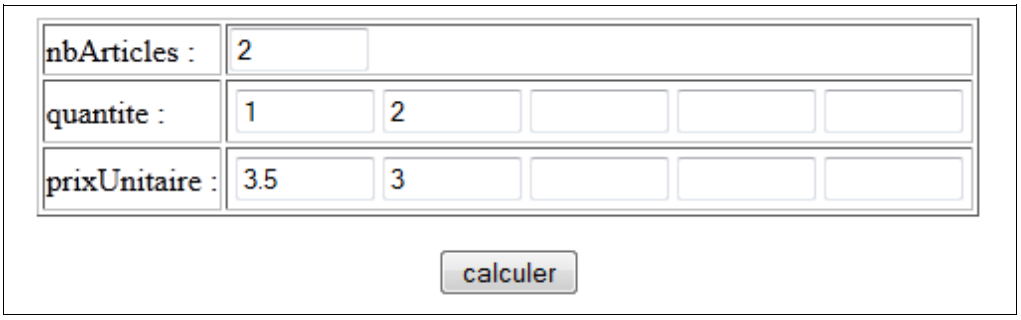


Exemple 2 : achat de plusieurs articles

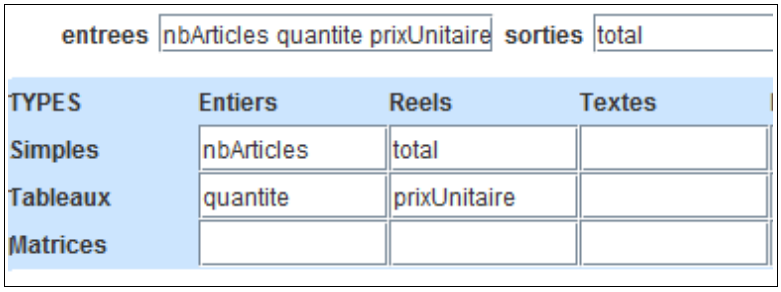
Le programme Javascript obtenu avec le traducteur est, là encore, opérationnel. Quand on l'exécute, on s'aperçoit que seul le nombre d'articles est saisi dans le formulaire html, les autres informations (prix unitaire et nombre d'exemplaires pour chaque article) étant saisies une à une :



Il serait plus convivial pour l'utilisateur de regrouper toutes les saisies dans le formulaire :



Il est possible d'obtenir le code de ce formulaire en complétant l'onglet Principal, puis en cliquant sur le bouton **Nouveau** :



PluriAlgo génère alors un code (compréhensible uniquement par des lecteurs avertis), dans lequel les données saisies par l'utilisateur sont stockées dans deux tableaux (quantite et prixUnitaire).

Mais pour compléter ce code, il suffit heureusement d'être suffisamment perspicace pour déterminer l'endroit où il faut le faire, c'est à dire juste avant l'instruction affichant le total :

```
var MAX_TAB = 10; // taille maximale des tableaux
// programme principal
function main (formu) {
      quantite = Array(MAX_TAB);
      prixUnitaire = Array(MAX_TAB);
      nbArticles = eval( formu["zone_nbArticles"].value );
      for(i1=0; i1<nbArticles; i1++) {
            quantite[i1] = eval( formu["zone_quantite_" +
      }
      for(i1=0; i1<nbArticles; i1++) {
            prixUnitaire[i1] = eval( formu["zone_prixUnita:
      }
      total=0;
      for(i1=0; i1<nbArticles; i1++) {
            total=total + quantite[i1]*prixUnitaire[i1];
      }
      document.writeln("total : "); document.writeln(tota:
      document.close();
}
```

Comme il s'agit d'une sommation, il est possible d'ajouter ces instructions en utilisant l'onglet Boucles :