

Você, cliente da plataforma Virtual Target, certamente já deve ter ouvido falar a respeito da programação BASIC em templates de email marketing. Mas, mesmo que não a conheça ou não saiba como utilizá-la, este manual pretende explicar, passo a passo, o uso da programação BASIC para criar templates personalizados, direcionados a diferentes perfis de público.

Em linhas gerais, a programação BASIC permite a criação de conteúdos específicos para determinados perfis de público, dentro de um único template de email marketing. Num exemplo prático, que será ilustrado mais adiante neste documento, se uma base de contatos possui um campo "Sexo" com os atributos "Masculino" e "Feminino" definidos, é possível criar um template que contemple conteúdos para estes dois tipos de público. Assim, com algumas linhas de programação BASIC, é possível indicar, dentro de apenas um template, quais seções e/ou conteúdos da mensagem serão modificados de acordo com cada perfil que a receber.

Por que utilizar a programação BASIC para email marketing:

BASIC permite o desenvolvimento de templates personalizados

A personalização de templates com programação BASIC vai muito além da inserção de chamadas para campos da base de contatos, como o nome/sobrenome dos contatos na saudação das mensagens. Com a programação BASIC, é possível determinar o conteúdo que será exibido na mensagem se o perfil do contato for X ou Y. O que vai determinar este "X" ou "Y" é a própria configuração da base de contatos: quanto mais rica em informações de cada indivíduo, maiores serão as possibilidades de uso da personalização com BASIC.

BASIC otimiza a produção de templates e reduz o tempo de desenvolvimento;

Na rotina de produção de templates para email marketing, é muito comum que os designers/desenvolvedores criem diferentes templates que serão enviados a certos perfis de público. Uma empresa que deseja enviar uma mensagem para seus clientes do sexo feminino e outra mensagem para os clientes do sexo masculino, não precisa criar um template para cada um. Utilizando BASIC dentro do código HTML, é possível criar apenas um template, que contém as informações a serem enviadas aos dois públicos.

BASIC diminui a quantidade de envios realizados

Se o uso da programação BASIC reduz a quantidade de templates desenvolvidos, também reduz a quantidade de envios realizados, pois um envio está condicionado ao uso de apenas um template. Além disso, "condensar" as diferentes mensagens aos diferentes públicos dentro de um único template otimiza a análise dos relatórios. Utilizando o primeiro exemplo da empresa que deseja enviar uma mensagem para o público feminino e outra para o público masculino, ao invés de realizar dois envios (um para cada

template), utilizando BASIC será necessário apenas um envio. Se considerarmos que cada envio oferece mais de 50 relatórios para analisar, quanto menos envios uma campanha tiver, mais fácil será a análise dos resultados.

BASIC é fácil de aplicar

Em suma, a programação BASIC é apenas uma forma de apontar diferentes blocos de código HTML a serem exibidos para certos perfis de público, que são identificados através dos campos adicionais configurados na Virtual Target. Como estes campos já estão cadastrados na plataforma, não há necessidade de declará-los como variáveis no início de cada trecho de código BASIC.

Onde utilizar a programação BASIC?

É possível inserir trechos de programação BASIC no Assunto (Subject) da mensagem e em qualquer lugar do código HTML do template, personalizando desde o título da página web, saudação, textos, imagens e até links de páginas de destino.

Basta identificar o trecho do código HTML que deseja personalizar e, imediatamente antes da primeira tag deste trecho, iniciar a programação BASIC. Da mesma forma, deve-se inserir o fechamento de BASIC imediatamente após a última tag do trecho. A estrutura é a seguinte:

```
<% IF condição THEN %>  
    <p>Bloco em HTML a ser personalizado</p>  
<% END IF %>
```

Assim, o conjunto de símbolos <% (menor + porcentagem) marca o início da programação BASIC, e o conjunto de símbolos %> (porcentagem + maior) marca o fim de um trecho programado com BASIC. Também é possível inserir comentários dentro de um bloco BASIC. Ele deve iniciar com o caractere # e ter uma linha reservada. Veja os exemplos:

ERRADO:

```
<% IF @UF = "SP" THEN # Valida se o estado é São Paulo %>
```

CORRETO:

```
<% IF @UF = "SP" THEN  
    # Valida se o estado é São Paulo  
>
```

Como em qualquer outra linguagem de programação, BASIC também utiliza instruções condicionais, funções e operadores matemáticos. Se você é cliente da Virtual Target e deseja consultar as instruções detalhadas de todas as funções e operadores da programação BASIC, entre em contato com a equipe de Suporte Técnico e solicite o **Manual Completo de Programação BASIC**.

Como utilizar a programação BASIC?

Para ilustrar o uso da programação BASIC em templates de email marketing, vamos tomar como exemplo a seguinte base de contatos:

Nome	E-mail	Sexo	Total de compras	Interesse
John	john@gmail.com	Masculino	500	Negócios
Mark	mark@provedor.com.br	Masculino	1000	Marketing
Anne	anne@hotmail.com	Feminino	2500	Informática

Na plataforma Virtual Target, os campos "nome", "email", "nascimento" e "celular" são considerados campos padrão e, por isso, configurados automaticamente no momento do upload da base, cada qual com as variáveis: @NOME, @EMAIL, @NASCIMENTO e @CELULAR. Neste exemplo, utilizamos apenas dois dos campos padrão: Nome e Email. Os demais devem ser cadastrados como Campos Adicionais. Para este tutorial, vamos pressupor que os campos adicionais "sexo", "total de compras" e "interesse" já foram devidamente configurados na Virtual Target com as variáveis: @SEXO, @TCOMPRAS, @INTERESSE.

Personalizando o conteúdo da mensagem com BASIC

Uma das aplicações de BASIC, já mencionada anteriormente, é a personalização da mensagem utilizando, por exemplo, o campo "sexo", porém, pode-se utilizar quaisquer outros campos existentes na base de contatos, desde que cadastrados na Virtual Target como Campos Adicionais. Vamos ilustrar, a seguir, o uso da programação BASIC para construir um texto de saudação apropriado para cada sexo:

Exemplo 1

```
<html>
  <head>
    <title>Exemplo de BASIC 1</title>
  </head>
  <body>
    <%IF @SEXO= "Masculino" THEN %>
<!--todo indivíduo da base de contatos que tiver o campo SEXO definido como Masculino, receberá a mensagem a seguir!-->
      <p>Prezado @NOME,</p>
    <% ELSE %>
```

```

        <% IF @SEXO= "Feminino" THEN %>
<!--todo indivíduo da base de contatos que tiver o campo SEXO definido como
Feminino, receberá a mensagem a seguir:-->
        <p>Prezada @NOME,</p>
        <% ELSE %>
<!--todo indivíduo da base de contatos que tiver o campo SEXO definido com
qualquer outra informação, receberá a mensagem a seguir:-->
        <p>Prezado(a) @NOME,</p>
        <% END IF %>
    <% END IF %>
</body>
</html>

```

Note que, em nossa base de contatos fictícia, temos cadastrados um indivíduo do sexo masculino (John), um do sexo feminino (Anne) e um sem informação de sexo cadastrada (Mark). Assim, de acordo com o código do Exemplo 1:

1) John receberá em seu email a seguinte saudação:

Prezado John,

2) Anne receberá em seu email a saudação:

Prezada Anne,

3) Mark, como foi cadastrado sem qualquer informação sobre sexo, receberá:

Prezado(a) Mark,

Personalizando o Assunto (Subject) da mensagem com BASIC

Além do corpo da mensagem, também é possível utilizar a programação BASIC para compor o Assunto (Subject). Vamos exemplificar, a seguir, o uso da programação BASIC para compor um assunto de mensagem diferente para cada perfil, de acordo com seu total de compras cadastrado na base:

Exemplo 2

```

<%
    SET @CARTAO = ""
    #Declaração da variável @CARTAO com valor vazio.
    IF @TCOMPRAS <= "500" THEN
    #Se o Total de Compras do destinatário for menor ou igual a 500,
        SET @CARTAO = "Bronze"
    #o tipo de cartão exibido no Assunto será Bronze.
    END IF
    IF @TCOMPRAS >= "2000" THEN
    #Se o Total de Compras do destinatário for maior ou igual a 2000,
        SET @CARTAO = "Ouro"
    #o tipo de cartão exibido no Assunto será Ouro.
    END IF

```

```
IF @CARTAO = "" THEN
#Se o Total de Compras do destinatário for diferente de quaisquer
valores definidos anteriormente, isto é, for maior que 500, porém, menor que
2000,
    SET @CARTAO = "Prata"
#o tipo de cartão exibido no Assunto será Prata.
END IF
%>
```

@NOME, retire seu Cartão @CARTAO em nossa loja

No Exemplo 2, a proposta é exibir um Assunto (Subject) diferente para cada indivíduo conforme seu total de compras realizadas. Assim, de acordo com nossa base fictícia e a programação BASIC utilizada, teremos:

John, cujo Total de Compras é inferior a 500, receberá a mensagem com o seguinte Assunto ou Subject: **John, retire seu Cartão Bronze em nossa loja**

Mark, cujo Total de Compras é maior do que 500, porém, menor do que 2000, receberá a mensagem com o seguinte Assunto ou Subject: **Mark, retire seu Cartão Prata em nossa loja**

Anne, cujo Total de Compras é superior a 2000, receberá a mensagem com o seguinte Assunto ou Subject: **Anne, retire seu Cartão Ouro em nossa loja**

Note que, no Exemplo 2, toda a programação BASIC é inserida antes do texto do Assunto, pois ao longo do código a variável @CARTAO tem seu valor alterado conforme as condições do Total de Compras. Se a linha do assunto fosse inserida antes do bloco de BASIC (quando a variável @CARTAO ainda não foi declarada, o texto exibido para os contatos apresentaria um espaço em branco na frase, como neste exemplo: **John, retire seu Cartão em nossa loja.**

Inserindo a data de recebimento da mensagem no template

Além dos campos utilizados na base de contatos e cadastrados na Virtual Target, também é possível construir instruções em BASIC utilizando outras variáveis, tais como data e tempo. Assim, é possível exibir, no template, a data/hora exatas em que o destinatário recebeu a mensagem.

Este recurso pode ser muito útil em mensagens cuja lógica do conteúdo esteja relacionada à data do recebimento, tais como newsletters, emails promocionais e transacionais. Como a data exibida no template é fornecida pelo servidor da Virtual Target no momento da entrega da mensagem, ela constitui uma informação que não pode ser alterada.

Newsletters: a data reforça o caráter periódico e atualizado da mensagem.

Emails promocionais: facilita a comparação entre a data do recebimento da mensagem e a data de validade de determinada promoção divulgada, especialmente nos casos de cupons promocionais que devem ser impressos e apresentados em lojas físicas.

Emails transacionais: permite que o destinatário perceba o tempo de resposta de uma empresa para determinada ação realizada no website, como confirmações de compra, prazos para entregas, respostas a dúvidas técnicas e mensagens utilizadas no double opt-in.

A seguir, veja o código necessário para exibir uma data por extenso, no formato: São Paulo, (dia) de (mês) de (ano). (Dia da Semana).

Exemplo 3

```
<%  
  IF @WDAY = "1" THEN  
    SET @NOMEDIA = "Domingo"  
  END IF  
  IF @WDAY = "2" THEN  
    SET @NOMEDIA = "Segunda-feira"  
  END IF  
  IF @WDAY = "3" THEN  
    SET @NOMEDIA = "Terça-feira"  
  END IF  
  IF @WDAY = "4" THEN  
    SET @NOMEDIA = "Quarta-feira"  
  END IF  
  IF @WDAY = "5" THEN  
    SET @NOMEDIA = "Quinta-feira"  
  END IF  
  IF @WDAY = "6" THEN  
    SET @NOMEDIA = "Sexta-feira"  
  END IF  
  IF @WDAY = "7" THEN  
    SET @NOMEDIA = "Sábado"  
  END IF  
  IF @MON = "1" THEN  
    SET @NOMEMES = "janeiro"  
  END IF  
  IF @MON = "2" THEN
```

```
    SET @NOMEMES = "fevereiro"
END IF
IF @MON = "3" THEN
    SET @NOMEMES = "março"
END IF
IF @MON = "4" THEN
    SET @NOMEMES = "abril"
END IF
IF @MON = "5" THEN
    SET @NOMEMES = "maio"
END IF
IF @MON = "6" THEN
    SET @NOMEMES = "junho"
END IF
IF @MON = "7" THEN
    SET @NOMEMES = "julho"
END IF
IF @MON = "8" THEN
    SET @NOMEMES = "agosto"
END IF
IF @MON = "9" THEN
    SET @NOMEMES = "setembro"
END IF
IF @MON = "10" THEN
    SET @NOMEMES = "outubro"
END IF
IF @MON = "11" THEN
    SET @NOMEMES = "novembro"
END IF
IF @MON = "12" THEN
    SET @NOMEMES = "dezembro"
END IF
%>
```

```
<html>
<head>
<title></title>
</head>
<body>
<p>São Paulo, @DAY de @NOMEMES de @YEAR. @NOMEDIA.</p>
</body>
</html>
```

Supondo que a mensagem do Exemplo 3 tenha sido recebida pelo destinatário no dia 26/05/2008, o código HTML produzirá o seguinte resultado no template:
São Paulo, 25 de maio de 2008. Segunda-feira.

Considerações Finais

Se você é cliente da Virtual Target e deseja compreender melhor a sintaxe da programação BASIC, entre em contato com a equipe de Suporte Técnico e solicite o Manual Completo de Programação BASIC. Utilize o Manual e os exemplos deste documento para enriquecer suas mensagens de email marketing e otimizar a produção de templates.